

**PERENCANAAN DATABASE QUERY DENGAN MANIPULASI ALJABAR
RELASIONAL**



*Diajukan Untuk Memenuhi Salah Satu Syarat Meraih Gelar Sarjana Matematika
Jurusan Matematika pada Fakultas Sains dan Teknologi
Universitas Islam Negeri (UIN) Alauddin Makassar*

Oleh

FIRMANSYAH SALAM

60600114056

**JURUSAN MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI (UIN) ALAUDDIN
MAKASSAR
2019**

PERNYATAAN KEASLIAN SKRIPSI

Dengan penuh kesadaran, penyusun yang bertanda tangan dibawah ini dinyatakan bahwa skripsi ini benar adalah hasil karya penyusun sendiri. Jika di kemudian hari terbukti bahwa ia merupakan duplikat, tiruan, plagiat, atau dibuat oleh orang lain, sebagian atau seluruhnya, maka skripsi dan gelar yang diperoleh karenanya batal demi hukum.

Makassar, Maret 2019

Penyusun,

FIRMANSYAH SALAM

NIM: 60600114056

UNIVERSITAS ISLAM NEGERI
ALAUDDIN
M A K A S S A R

PENGESAHAN SKRIPSI

Skripsi yang berjudul “Perencanaab Database Query dengan Manipulasi Aljabar Relasional”, yang disusun oleh Saudara **Firmansyah Salam**, Nim: **60600114056** Mahasiswa Jurusan Matematika pada Fakultas Sains dan Teknologi Universitas Islam Negeri (UIN) Alauddin Makassar, telah diuji dan dipertahankan dalam sidang *munaqasyah* yang diselenggarakan pada hari Selasa tanggal **06 Agustus 2918 M**, bertepatan dengan **05 Dzulhijjah 1440 H**, dinyatakan telah dapat diterima sebagai salah satu syarat untuk memperoleh gelar Sarjana Matematika (S.Mat).

Makassar, 06 Agustus 2918 M
05 Dzulhijjah 1440 H

DEWAN PENGUJI

Ketua : Prof. Dr. H. Arifuddin, M.Ag.

Sekretaris : Ermawati, S.Pd., M.Si.

Munaqisy I : Muh. Irwan, S.Si., M.Si.

Munaqisy II : Dr. Hasyim Haddade, S.Ag., M.Ag.

Pembimbing I : Risnawati Ibtnas, S.Si., M.Si.

Pembimbing II : Adnan Sauddin, S.Pd., M.Si.

(.....)
(.....)
(.....)
(.....)
(.....)
(.....)

Diketahui oleh:

Dekan Fakultas Sains dan Teknologi
UIN Alauddin Makassar

Prof. Dr. H. Arifuddin, M.Ag
Nip. 19691205/199303 1 001

Motto Dan Persembahan

Motto

“Berhenti Menunda Dan Mulai Melakukan, Karena Waktu Tidak Akan Menunggumu Untuk Memulai”

Kupersembahkan Tugas Akhir Ini Kepada:

“Karya ini kupersembahkan kepada Rabb-ku, Allah swt. yang telah memberiku semangat dan harapan dibalik keputusasaan”

“Persembahan kecil untuk kedua orang tuaku, Salam Nur dan Nursyam. yang tidak pernah berhenti memberi semangat, doa, dorongan, nasehat dan kasih sayang serta pengorbanan demi kesuksesanku.”

“”

“kepada seluruh warga matematika universitas islam negeri”

“Almamater UIN Alauddin Makassar”

UNIVERSITAS ISLAM NEGERI
ALAUDDIN
M A K A S S A R

KATA PENGANTAR

Assalamualaikum Wr.Wb

Syukur Alhamdulillah atas kesehatan, kesempatan dan kenikmatan yang telah Allah Swt karuniakan, atas segala Inayah, Taufiq dan Hidayah-Nya , sehingga dengan izin-Nya lah sehingga penulis dapat menyelesaikan skripsi yang berjudul **“Perencanaan Database Query Dengan Manipulasi Aljabar Relasional”**. Banyak kendala dan hambatan yang dilalui oleh penulis dalam penyusunan Skripsi ini, akan tetapi dengan segala usaha yang penyusun lakukan sehingga semuanya itu dapat teratasi.

Salam dan salawat kita hanturkan selalu kepada baginda Rasulullah Muhammad Saw, sebagai nabi penutup para nabi beserta keluarganya, sahabatnya, dan orang-orang yang mendakwahkan risalah-Nya dan yang telah membimbing umat ke jalan lurus, serta orang – orang yang berjihad di jalan-Nya hingga akhir zaman.

Skripsi ini merupakan salah satu syarat yang harus ditempuh oleh mahasiswa Fakultas Sains dan Teknologi, Universitas Islam Negeri Alauddin Makassar untuk meraih gelar Sarjana S-1(Sarjana Matematika).

Dengan menyelesaikan Skripsi ini penulis tidak dapat melakukan sendiri melainkan berkat bantuan dari berbagai pihak. Oleh karena itu dengan segenap ketulusan hati penulis mengucapkan terima kasih sedalam-dalamnya kepada:

Allah SWT yang telah melimpahkan Rahmat dan KaruniaNya sehingga skripsi ini dapat terselesaikan.

Ayahanda tercinta Salam Nur, Ibunda tercinta Nursyam, dan Keluarga Besar yang telah memberikan doa, dorongan moral dan material serta perhatian dan kasih sayang yang diberikan kepada penulis dalam menyelesaikan skripsi ini.

1. Bapak Prof. Dr. Musafir Pabbari, M.S., Rektor Universitas Islam Negeri Alauddin Makassar.
2. Bapak Prof. Dr. H. Arifuddin Ahmad, M.Ag., Dekan Fakultas Sains dan Teknologi, Universitas Islam Negeri Alauddin Makassar.
3. Bapak Irwan, S.Si., M.Si., Ketua Jurusan Matematika Fakultas Sains dan Teknologi, Universitas Islam Negeri Alauddin Makassar..
4. Ibu Risna Wati Ibbas, S.Si., M.Si Pembimbing I yang telah meluangkan waktu dalam membimbing, memberi saran dan kritikan untuk kesempurnaan penyusunan skripsi ini.
5. Adnan Sauddin, S.Sd., M.Si Pembimbing II yang telah meluangkan waktu dalam membimbing, memberi saran dan kritikan untuk kesempurnaan penyusunan skripsi ini.
6. Muh. Irwan S.Si., M.Si Penguji I yang telah meluangkan waktu dalam membimbing, memberi saran dan kritikan untuk kesempurnaan penyusunan skripsi ini.
7. Dr. Hasyim Haddade, M. Ag, Penguji II yang telah meluangkan waktu dalam menguji, memberi saran dan kritikan untuk kesempurnaan penyusunan skripsi ini
8. Bapak / Ibu Dosen Para Staf dan Pengajar Fakultas Sains dan Teknologi, Universitas Islam Negeri Alauddin Makassar, yang telah memberikan doa dan

dorongan moral serta perhatian dan kasih sayang yang diberikan kepada penulis dalam menyelesaikan skripsi ini.

9. Samriati, S.Si, dan Nurul Wulandari S.Mat, Operator Jurusan Matematikayang telah meluangkan waktu dalam membantu pengurusan berkas kelengkapan ujian.

10. Semua pihak yang telah membantu dalam penyelesaian skripsi ini.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna, oleh karena itu, kritik dan saran yang bersifat membangun untuk kesempurnaan skripsi ini sangat diharapkan. Akhir kata, penulis berharap semoga Allah swt, Membalas segala kebaikan semua pihak yang telah membantu dalam penyelesaian skripsi ini, Semoga skripsi ini membawa manfaat bagi kita semua dan terutama pengemban ilmu pengetahuan.

Gowa, Maret 2019

Penulis

Firmansyah Salam

60600114056

DAFTAR ISI

HALAMAN JUDUL	i
PERNYATAAN KEASLIAN SKRIPSI.....	ii
PENGESAHAN SKRIPSI.....	iii
MOTTO DAN PERSEMBAHAN	iv
KATA PENGANTAR.....	v-vii
DAFTAR ISI.....	viii-
ix	
DAFTAR TABEL	x-xi
DAFTAR SIMBOL	xii
ABSTRAK	xiii
BAB I PENDAHULUAN.....	1-7
A. Latar Belakang	1
B. Rumusan Masalah.....	5
C. Tujuan Penelitian	5
D. Manfaat Penelitian	5
E. Batasan Masalah	6
F. Sistematika Penulisan	6
BAB II TINJAUAN PUSTAKA	8-30
A. Aljabar	8
B. Aljabar Relasional.....	9
C. Basis Data	10
D. Objek Tabel.....	11
E. Notasi Aljabar Relasional	13
F. Sistem Basis Data	16
G. Perencanaan Basis Data	19
H. Fungsi.....	21
I. Relasi Menggunakan SQL	24
J. Bahasa SQL	27

K. Query Sebagai Ekpresi Aljabar Relasional.....	30
BAB III METODOLOGI PENELITIAN	31
A. Jenis Penelitian	31
B. Tempat dan Waktu Penelitian.....	31
C. Prosedur Penelitian	31
BAB IV HASIL DAN PEMBAHASAN	32-59
A. Hasil.....	32
B. Pembahasan	56
BAB V KESIMPULAN DAN SARAN	60
A. Kesimpulan.....	60
B. Saran.....	60
DAFTAR PUSTAKA.....	xiv
LAMPIRAN	
RIWAYAT PENULIS	

DAFTAR TABEL

Tabel 2.1 Data karyawan

Tabel 2.2 tipe data yang paling sering di gunakan

Tabel 2.3 Students S1 sebelum pembaruan

Tabel 2.4 Students S1 setelah pembaruan

Tabel 4.1 Struktur *Database* Biblio

Tabel 4.2 Struktur *Database* Biblio_Author

Tabel 4.3 Table biblio

Tabel 4.4 Hasil dari menampilkan data tanpa syarat

Tabel 4.5 Hasil dari menampilkan data bersyarat

Tabel 4.6 Tabel biblio pada pengujian *Projection*

Tabel 4.7 Hasil Pengujian *Projection* tanpa bersyarat

Tabel 4.8 Hasil Pengujian *Projection* bersyarat

Tabel 4.9 Table Biblio Pada pengujian *Union*

Tabel 4.10 Table Biblio_author Pada pengujian *Union*

Tabel 4.11 Hasil Pengujian *Union* Bersyarat

Tabel 4.12 Tabel Biblio pada pengujian *insert*

Tabel 4.13 Hasil Pengujian *Insert*

Tabel 4.14 Tabel biblio Pada pengujian *Delete*

Tabel 4.15 Hasil Pengujian *Delete* tanpa bersyarat

Tabel 4.16 Hasil Pengujian *Delete* bersyarat

Tabel 4.17 Tabel Biblio Pada pengujian *update*

Tabel 4.18 Pengujian *Update* secara keseluruhan

Tabel 4.19 Pengujian *Update* Bersyarat

Tabel 4.20 Jenis tabel dalam *database* Slims

Tabel 4.21 Tabel pengujian *Rename*

Tabel 4.22 Tabel biblio Pada pengujian *Rename*

Tabel 4.23 Hasil pengujian *Rename*

Tabel 4.24 Jenis tabel dalam *database* Slims

Tabel 4.25 Hasil penggunaan *Drop* pada database

Tabel 4.26 jenis tabel pada *database*

Tabel 4.27 Hasil penambahan tabel pada *database*



DAFTAR SIMBOL

σ : Menambahkan Data

π : *Projection*

\cup : *Union*

\cup : *Insert*

$-$: *Delete*

δ : *Update*

ρ : *Rename*

$-$: *Drop*

$+$: Menambahkan tupel



ABSTRAK

Nama Penyusun : Firmansyah Salam

NIM : 60600114056

Judul : Perencanaan Database Query Dengan Manipulasi Aljabar Relasional

Aljabar Relasional adalah bahasa *query* formal yang terkait dengan model relasional. *Query* dalam aljabar disusun dengan menggunakan satu kumpulan operator. Dalam pembuatan database tidak lepas dari fungsi dan relasi himpunan di mana pada pembuatan tabel-tabel data yang dikelola akan saling berkaitan dari tabel satu ke tabel lainnya, penelitian ini bertujuan untuk mengetahui hasil penerepan database Query dengan menggunakan aljabar Relasional. Berdasarkan hasil penelitian pada pengolahan database aljabar relasional terbagi atas beberapa perintah yaitu Data Manipulation Language (DML) yang dimana mencakup menampilkan data, projection, union, insert, delete, dan update. Data Definition Language (DDL) yang dimana mencakup rename, drop dan menambahkan tabel.

Kata Kunci : *Database, Aljabar relasional, query relasional*



BAB I

PENDAHULUAN

A. Latar Belakang

Perkembangan database tidak lepas dari perkembangan perangkat keras dan perangkat lunak, Ketiga Komponen tersebut merupakan komponen pembentuk *system database*. *System database* yang ada sampai sekarang, hanya mampu menangani data yang bersifat pasti. Begitu pula pada proses query yang menggunakan bahasa *Structure Query Language* (SQL), kondisi-kondisi yang diberikan hanya mampu menangani kondisi yang bersifat pasti. SQL merupakan standar yang digunakan untuk mengakses, manipulasi dan mengontrol database bertujuan untuk skalabilitas, reabilitas dan konsistensi data.

Dalam mengelolah database yang harus di perhatikan apakah database tersebut sudah normal atau belum normal, dimana tujuan normalisasi database adalah untuk menghilangkan dan mengurangi redudansi data serta memastikan dependensi data (data berada pada tabel yang tepat) sehingga tabel-tabel tersebut terkait secara *logic*. Jika data dalam database tersebut belum di normalisasi maka akan terjadi 3 kemungkinan yang akan merugikan sistem secara keseluruhan yaitu terjadinya anomali *insert, delete, dan update* pada data yang berakibatkan inkonsistensi.

Dalam pembuatan database tidak lepas dari fungsi dan relasi himpunan di mana pada pembuatan tabel-tabel data yang dikelola akan saling berkaitan dari tabel satu ke tabel lainnya dalam matematika fungsi adalah relasi khusus

yang memasangkan setiap anggota himpunan (domain) dengan tepat satu anggota ke himpunan lainnya (kodomain), relasi dalam matematika hubungan antara anggota suatu himpunan (domain) dengan himpunan lainnya (kodomain). Di mana hubungan antara data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. Dalam pembuatan database bukan suatu hal sulit lagi di karenakan perkembangan teknologi yang begitu pesat salah satunya program komputer yang di gunakan dalam mengelolah komputer, di mana dalam pengelolaan database yang paling sering di gunakan Query atau biasa di sebut perintah-perintah untuk mengakses data pada database atau basis data.

Bagian dari ilmu komputer yang di gunakan adalah aljabar relasional, cabang dari logika predikat tingkat pertama dan aljabar himpunan, yang menangani suatu set relasi hingga yang memiliki sifat tertutupan dengan operator-operator tertentu. Operator ini bertindak dengan satu atau lebih relasi untuk menghasilkan suatu relasi (gabungan), Aljabar relasional adalah sebuah bahasa query prosedural yang terdiri dari sekumpulan operasi dimana masukannya adalah satu atau dua relasi dan keluarannya adalah sebuah relasi baru sebagai hasil dari operasi tersebut. Operasi-operasi dasar dalam aljabar relasional adalah : *selection*, *project*, *union*, *insert*, *delete*, dan *update*. Disamping operasi-operasi dasar terdapat beberapa operasi tambahan seperti *rename* dan *drop*

Adapun ayat yang berhubungan dengan penelitian ini adalah surah Shad ayat 27

وَمَا خَلَقْنَا السَّمَاءَ وَالْأَرْضَ وَمَا بَيْنَهُمَا بَطْلًا ذَٰلِكَ ظَنُّ الَّذِينَ كَفَرُوا فَوَيْلٌ لِلَّذِينَ
كَفَرُوا مِنَ النَّارِ ﴿٢٧﴾

Terjemahnya :

Dan kami tidak menciptakan langit dan bumi dan apa yang ada di antara keduanya tanpa hikmah. Yang demikian itu adalah anggapan orang-orang kafir; maka celakalah orang-orang kafir itu, karena mereka akan masuk neraka.¹

Ayat ini menjelaskan Allah Swt. Menceritakan bahwa tidak sekali-kali dia menciptakan makhluknya dengan main-main, melainkan Dia ciptakan mereka supaya mereka menyembahnya. Kemudian Allah akan menghimpun mereka di hari perhimpunan, maka Dia akan memberi pahala kepada orang yang taat dan mengazab orang yang kafir. Karena itulah di sebutkan dalam firmanya dan Kami tidak akan menciptakan langit dan bumi dan apa yang ada di antara keduanya secara sia-sia. Yang demikian itu adalah anggapan orang-orang kafir, Yakni orang-orang yang tidak percaya kepada hari berbangkit dan tidak pula kepada hari kembali, melainkan hanya percaya kepada kehidupan di dunia ini saja. maka celakalah orang-orang kafir itu karena mereka akan masuk neraka, yang di maksud celakalah mereka di hari mereka kembali saat mereka dibangkitkan karena akan memasuki neraka yang telah disediakan buat mereka. Kemudian Allah menjelaskan, bahwa

¹ Kementrian agama RI. *Al-Quran dan Terjemahannya*. 2008

termasuk keadilan dan hikmah-Nya Dia tidak menyamakan antara orang-orang mukmin dan orang-orang kafir.²

Berdasarkan ayat ini menjelaskan bahwa segala perintah Allah SWT yang di jalankan di muka bumi dengan baik maka akan berahir dengan baik pula, begitupun sebaliknya apabila perintah Allah SWT tidak di jalankan dengan baik maka hasilnya akan tidak baik juga, begitu pula dengan pemrograman dalam mengelolah *database* jika dijalankan dengan baik dan benar maka akan menghasikan *database* yang baik dan benar dan apabila dikerjakan dengan tidak baik dan tidak benar maka hasilnya akan tidak baik pula.

Beberapa penelitian yang telah di lakukan menggunakan manipulasi ajabar pada *database* diantaranya Menurut Eko Darmanto (2015) Kesesuaian antara teori dalam aljabar relasi dengan implementasi ke dalam sintaks bahasa SQL terlihat dari hasil luaran eksekusi program. Aljabar relasional sangat membantu adanya dalam penggunaan sintaks bahasa SQL. Menurut Mulyadi S, Armin Lawi dan Andani Achmad (2017) operasional aljabar relasional pada sistem data dapat diterapkan. Data yang telah disimpan kedalam format XML dapat diakses menggunakan operasi relasional, terutama untuk operasi pencarian yang merupakan inti dari sebuah aplikasi database. Menurut Slamet Sudaryanto (2007) pada penelitian ini menggunakan teknik *heuristic*, optimasi query berhubungan dengan teknik-teknik yang digunakan oleh DMBS (*database* Manajemen Sistem) sangat membantu

² M.Quraish shibab. *Al-Misbah*. Lentera hati. 2003

untuk memperoleh query agar menghasilkan query dengan waktu minimum. Dalam penelitian-penelitian sebelumnya menyatakan bahwa Aljabar relasional sangat membantu adanya kemungkinan penggunaan sintaks bahasa SQL dalam sebuah program *Database*.

Penentuan sebuah tabel dikatakan baik, tentunya memiliki rancangan-rancangan yang harus dilakukan. Dimana rancangan tersebut terdiri dari perancangan konseptual, logis, dan secara fisik. Selain rancangan tersebut proses normalisasi akan sangat berperan penting untuk diajukan pada suatu database sehingga akan diperoleh hasil tabel yang dapat dikatakan stabil. Selain itu dari database tersebut akan dibuat suatu manipulasi aljabar dan dituangkan dalam SQL untuk relasi tabel.

B. Rumusan Masalah

Berdasarkan latar belakang di atas rumusan masalah adalah bagaimana Penerapan *database Query* dengan menggunakan aljabar Relasional?

C. Tujuan

Adapun tujuan berdasarkan rumusan masalah di atas adalah untuk mengetahui hasil penerepan *database Query* dengan menggunakan aljabar Relasional

D. Manfaat

Adapun Manfaat yang diperoleh adalah:

1. Bagi penulis

Penulisan skripsi ini dapat menambah wawasan penulis tentang pengolahan *database*

2. Bagi pembaca

Sebagai bahan studi tambahan ilmu bagi Pembaca, dan dapat di jadikan referensi sebagai landasan akan penggunaan atau pembuatan *database* dengan manipulasi aljabar relasional pada program SQL

3. Bagi Universitas

Penulisan skripsi ini dapat menambah koleksi bahan pustaka yang bermanfaat bagi Universitas Islam Negeri Alauddin Makassar pada umumnya dan mahasiswa jurusan Matematika Fakultas Sains dan Teknologi pada khususnya.

E. Batasan Masalah

Adapun batasan masalah yang di teliti oleh penulis adalah

1. Mengambil data buku pada ruang baca jurusan matematika UINAM
2. Menerapkan *database query* menggunakan manipulasi aljabar relasional

F. Sistem penyusunan

1. Bab I Pendahuluan

Bab ini berisi tentang latar belakang masalah, rumusan masalah, tujuan penelitian, manfaat penelitian dan sistematika penulisan.

2. Bab II Tinjauan Pustaka

Bab ini berisi informasi satu teori-teori yang membahas tentang, database, aljabar relasional, pemrograman, sebelumnya dipaparkan teori-teori yang mendasari masalah yang diteliti.

3. Bab III Metodologi Penelitian

Bab ini berisikan bagaimana penelitian akan dilaksanakan dan langkah-langkah dalam cara penyelesaian dari rumusan masalah.

4. Bab IV Hasil dan Pembahasan

Bab ini menguraikan bagaimana penyelesaian masalah dan metode yang digunakan dari penelitian tersebut.

5. Bab V Penutup

Bab ini berisikan tentang kesimpulan dan saran.

6. Daftar Pustaka

Daftar pustaka ini memuat referensi yang terkait dengan penelitian ini.

BAB II

TINJAUAN PUSTAKA

A. Aljabar

Aljabar adalah salah satu bagian dari bidang matematika yang luas, bersama-sama dengan teori bilangan, geometri dan analisis. Dalam bentuk paling umum, aljabar adalah ilmu yang mempelajari simbol-simbol matematika dan aturan untuk memanipulasi simbol-simbol ini, aljabar adalah pemersatu dari hampir semua bidang matematika. Selain itu, aljabar juga meliputi segala sesuatu dari dasar pemecahan persamaan untuk mempelajari abstraksi seperti grup, gelanggang, dan medan. Aljabar elementer umumnya dianggap penting untuk setiap studi matematika, ilmu pengetahuan, atau teknik, serta aplikasi dalam kesehatan dan ekonomi. Aljabar abstrak merupakan topik utama dalam matematika tingkat lanjut, yang dipelajari terutama oleh para profesional dan pakar matematika.

Dalam aljabar, beberapa bilangan yang digunakan mungkin di ketahui tetapi bilangan-bilangan lainnya tidak diketahui atau tidak ditentukan, lebih tepatnya bilangan-bilangan tersebut dilambangkan dengan huruf. Sebagai contoh, ubahlah h jam dan m menit menjadi menit. Ini dilakukan dengan mengalihkan h dengan 60 dan hasilnya ditambah dengan m jadi, $h \cdot 60 + m = 60h + m$ sebagai suatu pernyataan aljabar. Karena pernyataan aljabar melibat bilangan-bilangan, maka bilangan-bilangan ini dapat dijumlahkan, dikurangi, dan sebagainya, mengikuti hukum – hukum yang sama yang berlaku pada operasi-operasi ini untuk bilangan yang diketahui. Sebagai contoh, hasil penjumlahan dari $5 \times 60 + 35$ dan $2 \times 60 + 35$ adalah $(5 + 2) \times 60 + 2 \times$

35 dengan cara yang sama, hasil penjumlahan dari $h \cdot 60 + m$ dan $k \cdot 60 + m$ adalah $(h + k) \cdot 60 + 2m$.³

B. Aljabar Relasional

Aljabar Relasional adalah salah satu dari dua bahasa *query* formal yang terkait dengan model relasional. *Query* dalam aljabar disusun dengan menggunakan satu kumpulan operator, *property* pokok adalah setiap operator dalam aljabar menerima (satu atau dua) contoh relasi sebagai argument dan menghasilkan contoh relasi. Property ini memudahkan operator *compose* membentuk satu *query* kompleks-ekspresi aljabar relasional secara berulang ditentukan untuk menjadi suatu relasi, satu operator aljabar unary berlaku untuk ekspresi tunggal, atau operator operator aljabar biner berlaku untuk dua ekspresi. Kita menjelaskan operator dasar aljabar (*selection*, *projection*, *union*, *cross-product*, dan *difference*), dan juga beberapa operator tambahan yang dapat dimasukkan ke dalam konteks operator dasar tetapi cukup sering muncul menuntut perhatian khusus, dalam bagian berikut.

Tiap aljabar relasional menggambarkan prosedur langkah-langkah untuk menghitung jawaban yang diinginkan, berdasarkan urutan penerapan operator di dalam *query*. Sifat aljabar yang *produral* membuat kita dapat melihat ekspresi aljabar sebagai satu rencana untuk mengevaluasi sebuah *query*, dan *system* Basis Data relasional sebenarnya menggunakan ekspresi aljabar untuk menyajikan rencana evaluasi *query*.⁴

³ Schidmit, Philip A dan Frank ayres. *Matematika universitas*. Erlangga. 2003

⁴ Gehrke, Johannes dan reghu ramakrishnan. *Sistem Manajemen Database*. Andi and McGraw-Hill Education. 2004.

C. Basis Data

Basis Data terdiri atas 2 kata, yaitu basis dan data. Basis dapat diartikan sebagai markas atau gudang, tempat bersarang atau berkumpul. Data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang direkam dalam bentuk angka, huruf symbol, teks, gambar, atau kombinasi. basis data sendiri dapat di definisikan sebagai himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.⁵

Entitas adalah objek dalam dunia nyata yang biasa dibedakan dari objek lain. Biasanya hal ini sangat membantu dalam mengidentifikasi sekumpulan entitas yang sama. Kumpulan tersebut dinamakan set entitas. Perhatikan bahwa beberapa set entitas tidak perlu dipisahkan, kumpulan dari *employee toy department* dan kumpulan *employee appliance departementt* dapat memiliki employee bernama john doe (yang kebetulan bekerja di kedua department tersebut).

Entitas dapat digambarkan dengan menggunakan sekumpulan atribut. Semua entitas yang ada pada satu set entitas tertentu memiliki atribut yang sama, inilah yang disebut serupa. (pernyataan ini adalah sebuah penyerderhanaan, atribut yang dipilih mencerminkan rincian yang akan di pakai untuk menyajikan informasi tentang entitas

⁵ Fathansyah. *Database. buku teks ilmu komputer*. Informasi. 1999

tersebut. Misalnya, set entitas *employees* bias menggunakan *name*, *social security number*, dan *parking lot* (lot) sebagai atribut. Dalam hal ini kita akan menyimpan *name*, *social security number (ssn)*, dan *parking lot* untuk setiap *employee*. Akan tetapi, kita tidak akan menyimpan alamat (atau jenis kelamin atau usia *employee*).⁶

D. Objek Tabel

Tabel merupakan container untuk menyimpan data yang memiliki tata letak yang sama. Jika data yang di buat adalah mengumpulkan informasi tentang karyawan, data tentang masing-masing karyawan biasanya akan memiliki keserupaan dan oleh sebab itu layak menjadi isi dari sebuah tabel yang betul-betul memuaskan.

Tabel terdiri atas baris (horizontal) dan kolom (vertikal). Dalam tabel yang *sensible* (diperlihatkan kembali di bawah ini) setiap baris berisi data tentang satu karyawan dan data yang terletak di dalam satu baris dikenal sebagai *record*.

Tabel 2.1 Data karyawan

<i>Employee No</i>	<i>First Name</i>	<i>Last Name</i>	<i>Date Of birth</i>	<i>Date Employed</i>
1	Many	Tomanny	12-Apr-66	1-May-99
2	Rosanne	Kolumn	21-Mar-77	1-Jan-00
3	Cas	Kade	1-May-77	1-Apr-02
4	norma	Lyzation	3-Apr-66	1-Apr-02

⁶ Gehrke ,Johannes dan reghu ramakrishnan. *Sistem Manajemen Database*. Andi and McGraw-Hill Education.2004.

5	juan	Tomani	12-Apr-66	1-Apr-02
---	------	--------	-----------	----------

Dalam membuat tabel di mana, tabel memiliki *field* dan *field* memiliki nama *field*. Namun demikian, jika lebih sering menggunakan *desing* ‘*desing view*’ maka akan ada permintaan lain selain nama *field*. Untuk setiap *field* pengguna akan diminta untuk memilih ‘tipe’ data yang akan disimpan oleh *field* (dikenal sebagai ‘data tipe’) dan *size* (ukuran) data akan ditempatkan di dalamnya.

Tipe-tipe data yang paling lazim digunakan ialah:

Tabel 2.2 tipe data yang paling sering di gunakan

<i>Text</i>	Sebagian besar karakter yang ada pada keyboard, termasuk angka; biasanya dibatasi hingga 255 karakter
<i>Memo</i>	Blok-blok besar berisi teks
<i>Number (atau numeric)</i>	Hanya bilangan, tanpa karakter teks
<i>Date/Time</i>	Seperti namanya; tanggal, waktu, atau keduanya
<i>currency</i>	Pada dasarnya bilangan numeric dengan empat tempat decimal dan sebuah lambing mata uang
<i>Auto number</i>	Nomor yang secara otomatis ditambahkan untuk setiap record baru
<i>Yes/No</i>	Untuk informasi terpisah yang bias dibagi ke dalam dua kategori; seperti True/false, Yes/No, up/down

<i>OLE Object</i>	Suatu objek seperti lembar-kerja Microsoft Excel, dokumen Microsoft Word, atau informasi gratis.
<i>Text</i>	Sebagian besar karakter yang apa pada keyboard, termasuk angka; biasanya dibatasi hingga 255 karakter

E. Notasi Aljabar relasional

- a) *Data Manipulation Language* (DML) adalah bahasa basis data yang bertujuan untuk mengakses dan memanipulasi data. Operasi-operasi pada DML meliputi.⁷

a) Menampilkan data (σ)

Digunakan untuk menyaring data berdasarkan suatu kriteria tertentu. Kriteria yang digunakan dapat digabung dengan logika *AND* (\wedge), *OR* (\vee) dan *NOT* (\neg). Selain logika juga terdapat operator pembandingan yang terdiri dari $=$, \neq , $<$, $>$, \leq dan \geq . Bentuk umumnya menampilkan data $\sigma_K(R)$, dimana K merupakan kondisi pembatas atau penyaring data, yang terdiri dari nama atribut, operator pembandingan dan nilai batasnya. Sedangkan R adalah satu atau lebih tabel yang digunakan. Dalam bahasa SQL seleksi ditempatkan pada klausa *WHERE* atau *HAVING* jika seleksinya setelah terjadi *GROUP BY*.

b) *Projection* (π)

Digunakan untuk memilih dan menampilkan atribut-atribut dari suatu tabel atau relasi. Misalkan terdapat $R = (A_1, A_2, \dots, A_n)$. Dimana R adalah relasi

⁷ Bobihu,yusran. *Pengantar Sistem Basis Data*. Alauddin University Press. 2013

atau tabel yang diperoleh dari entitas dan atau relationship. Sedangkan A_1, A_2, \dots, A_n Adalah atribut ke-1, sampai dengan atribut ke-n. Maka bentuk proyeksinya seperti pada notasi *projection* $\pi_{A_i}(R)$, dimana A_1 merupakan atribut-atribut dan datanya yang akan ditampilkan. Sedangkan R adalah satu atau lebih tabel yang digunakan. Pada sintaks bahasa SQL bentuk proyeksi dari aljabar relasional $\pi_{A_1, A_2, A_3}(R)$ adalah `SELECT A1, A2, A3 FROM R.`

c) *Union* (\cup)

Digunakan untuk menggabungkan dua atau lebih tabel yang memiliki kolom dengan jumlah yang sama. Misalkan terdapat relasi R dan S, maka operasi union adalah $R \cup S$. Dalam terjemahan bahasa SQL tabel R dan S, maka operasi *union* adalah $R \cup S$. terbentuk sintaksis `SELECT*FROM R UNION SELECT* FROM S`

d) *Insert* (\cup)

Digunakan untuk menambahkan data pada suatu tabel/relasi. Jika terdapat suatu tabel/relasi r dengan suatu ekspresi E yang berisi data-data yang akan dimasukkan pada tabel maka bentuk umumnya persamaan pada *insert* $r \leftarrow r \cup E$.

e) *Delete* ($-$)

Digunakan untuk menghapus suatu data tertentu pada suatu tabel. Jika terdapat suatu tabel atau relasi r dengan suatu kondisi ekspresi E yang akan

menentukan data mana yang akan dihapus, maka bentuk umumnya adalah $r \leftarrow r - E$.

f) *Update* (δ)

Menggunakan simbol Delta (δ) yang digunakan untuk merubah data. Jika terdapat tabel/relasi r dengan ekspresi perubahan data E pada suatu atribut A , maka bentuk umum *update* $\delta_A \leftarrow E(r)$.

2. *Data Definition Language* (DDL) adalah suatu perintah pada bahasa basis data, dalam hal ini menggunakan SQL, untuk mendefinisikan dan memodifikasi struktur data pada skema basis data.

a) *Rename* (ρ)

Digunakan untuk merubah nama tabel secara permanen dan sementara. Bentuk Aljabar relasi untuk merubah nama tabel memiliki persamaan. Perbedaan perubahan nama tabel secara permanen maupun sementara hanya terletak pada perintah SQL-nya. Bentuk umum dari perintah rename $\rho_{Newtable}(OldTable)$ Sebagai model diberikan contoh merubah nama tabel secara permanen tabel Coba menjadi Contoh. Maka aljabar relasionalnya adalah $\rho_{contoh}(coba)$ dengan sintaks bahasa SQL-nya `RENAME TABLE Coba TO Contoh.`

b) *Drop* ($-$)

Drop berfungsi untuk menghapus data yang telah disimpan dalam database. Dalam perintah *Drop* bertujuan menghapus seluruh tabel di dalam basis data.

c) Menambahkan tupel (+)

Dimana dalam pengolahan ini bertujuan menambahkan tabel baru pada basis data. Sebelum membuat tabel yang harus diketahui lebih dulu tipe data dan domain dari kolom/ *field* tabel yang akan di buat.

F. Sistem basis data

Sistem Basis data ini sudah bias dikembangkan, baik pada mesin-mesin komputer mikro sampai dengan komputer *main-frame* yang tujuan dari sitem tersebut secara keseluruhan adalah untuk melakukan perawatan informasi dan menyajikannya kapan saja dibutuhkan oleh pengguna. Sistem basis data terdiri atas 4 komponen penting.

1. Data

Tergantung konfigurasi sistemnya, maka data dalam basis data dapat merupakan data yang single-user (hanya satu pengguna yang beroperasi terhadap basis data) atau multi-user, di mana satu atau lebih user beroperasi secara bersama ke dalam basis data. Sehingga data dalam basis data terutama untuk system yang besar, harus terintegrasi (*integrated*) dan dapat dipakai bersama (*shared*). Pengertian terintegrasi bahwa sanya basis data dapat dipakai bersama (*shared*). Pengertian terintegrasi bahwasanya basis data dapat dipandang sebagai kumpulan berbagai file yang saling terhubung dan dengan sebagaian atau seluruh redundasi yang ada di antaranya dihilangkan. Data operasional yang dipakai oleh system

aplikasi dari suatu enterprise (organisasi, perusahaan, rumah sakit, universitas, departemen pemerintah, dan sebagainya).

2. Perangkat Keras

Perangkat keras yang dibutuhkan untuk manajemen basis data biasanya masih berupa mesin standar yang ada, dalam arti tidak ada kekhususan tertentu. Akan tetapi, karena sifatnya dalam access yang lebih sangat bervariasi (yang tentunya cenderung lebih banyak direct-access) maka suatu manajemen basis data akan lebih banyak membutuhkan media penyimpanan harddisk. Beberapa pabrik telah pula mencoba dengan apa yang dinamakan dengan “basis data mesin” yaitu suatu perangkat keras yang dirancang khusus untuk mendukung system basis data, namun tidak terlalu populer.

3. Perangkat Lunak

Antara fisik basis data (tempat dimana sesungguhnya suatu basis data tersimpan dalam media) dengan pengguna terdapat suatu piranti lunak yang disebut system manajemen basis data (DBMS) atau DB Manager. Semua kebutuhan access oleh pengguna, seperti pembentukan file (*create*), penambahan dat (*insert*), penghapusan (*delete/drop*), dan lain-lain, semua dilaksanakan oleh DBMS. Satu hal lagi, bahwa DBMS juga berfungsi untuk memberikan satu batas agar pengguna basis data tidak perlu memikirkan berbagai hal yang berkaitan dengan detail pada level perangkat-keras (Misalnya metode access)

4. Pengguna (*users*)

Ada 3 (tiga) klas pengguna basis data, termasuk di antaranya adalah berikut ini.

- a) Pengguna akhir (*end-user*), yaitu orang yang menggunakan data didalam basis data untuk kebutuhan tugas atau fungsinya. Pengguna ini dapat meng-access basis data secara on-line dengan memanfaatkan bahasa *query* (seperti SQL) atau secara on-line maupun proses batch menggunakan program aplikasi yang sudah dipersiapkan oleh programmer (biasanya dibimbing dengan menu), atau pula menggunakan program *utility* yang telah tersedia di dalam DBMS. Termasuk dalam kategori pengguna akhir adalah pemilik system (*enterprise*), para manajer, supervisor, operator (misalnya karyawan loket bank, bagian pembukuan), pelanggan, dan sebagainya yang terlibat langsung dalam penggunaan basis data menggunakan bahasa query seperti SQL. Atau pre-programmed SQL atau yang menggunakan alat bantu yang dikembangkan oleh programmer analis, ataupun browser.
- b) Programmer aplikasi, yaitu orang yang bertanggung jawab untuk menulis program aplikasi, biasanya menggunakan berbagai bahasa pemrograman, seperti cobol, pascal, PL/1, dan lain sebagainya.
- c) Administrator basis data (DBA), yaitu orang yang bertanggung jawab pada keseluruhan system basis data. DBA adalah orang atau sekelompok orang yang merupakan penanggung jawab pada penyelenggara basis data. Basis data administrator mempunyai fungsi yang meliputi berbagai kegiatan. Seperti

pengaturan penempatan data, pengamanan data, recovery prosedur, backup prosedur.⁸

G. Perancangan basis data

Permasalahan dalam perancangan basis data adalah bagaimana merancang struktur logika dan fisikal dari satu atau lebih database untuk memenuhi kebutuhan informasi yang diperlukan oleh pengguna sesuai dengan aplikasi-aplikasi yang ditentukan.

Perancangan basis data adalah proses untuk menentukan isi dan pengaturan data yang dibutuhkan untuk mendukung berbagai rancangan system. Tantangan dalam merancang *database* adalah bagaimana merancang sehingga *database* dapat memenuhi keperluan saat ini dan masa mendatang.

Proses Perancangan basis data dibagi menjadi 3 tahapan yaitu :

1. Perancangan basis data secara konseptual, tahapan ini merupakan upaya untuk membuat model yang masih bersifat konsep.
2. Perancangan basis data secara logis, merupakan tahapan untuk memetakan model konseptual ke model database yang akan dipakai (model relasional, hirarkis, atau jaringan).
3. Perancangan basis data secara fisis, merupakan tahapan untuk menuangkan perancangan database yang bersifat logis menjadi database fisis yang tersimpan

⁸ Subekti, Muhammad. system manajemen basis data. ghalia Indonesia. 2004. hal 1-5

pada media penyimpanan eksternal (yang spesifik terhadap DBMS yang dipakai)

Dalam melihat suatu sistem basis data, bila dikaitkan pada penggunaannya, maka pandangan (view) terhadap suatu basis data dapat dibagi dalam tiga tingkatan (level).

1. Pandangan Eksternal, merupakan pandangan individu masing-masing pengguna akhir (“end-user”: manajer enterprise, operator, programmer, dan lain-lain). Di level ini merupakan bentuk yang paling dekat dengan logika pandangan masing-masing pengguna terhadap data yang tersimpan dalam basis data. Pengguna pada level ini biasanya hanya berkepentingan pada sebagian (kecil) dari keseluruhan basis data karena masing-masing pengguna melihat data keseluruhan basis data karena masing-masing pengguna melihat data di dalam basis data sesuai porsi aplikasi yang menjadi tanggung jawab/wewenang saja.
2. Pandangan Konseptual, merupakan pandangan masyarakat basis data secara total keseluruhan, yaitu tingkat di mana pandangan masing-masing pengguna digabungkan menjadi satu konsep struktur dan terintegrasi sehingga merupakan gambaran dari keseluruhan basis data.
3. Pandangan Internal, merupakan pandangan penyimpanan mendekati pada tingkat pandangan basis data dalam penyimpanan fisik. Di sini diperlihatkan cara data tersimpan dalam media penyimpanan harddisk, misalnya. Meliputi berbagai hal seperti paging (blocking), clustering, indexing, dan lain sebagainya, yang sangat berorientasi pada perangkat keras komputer.

Struktur penyimpanan menjelaskan organisasi data di dalam penyimpanan sekunder dalam media *access* langsung (*direct access*) seperti harddisk, drum, dan sebagainya. Media *access sequential (tape)* tidak dibahas, mengingat banyaknya keterbatasannya untuk dipakai dalam system basis data. Media tape biasanya sangat berperan dalam *system backup(copy basis data)*. Dalam perencanaan basis data, struktur ini berkaitan erat dengan konseptual tingkat internal dari system basis data. operasi pengguna terhadap basis data dilakukan via DML dengan terminologi record secara eksternal (pandangan pengguna sendiri), pada gilirannya, maka DBMS mengkonversikan instruksi tersebut pada operasi internal atau record tersimpan. Untuk selanjutnya access method akan melakukan konversi dari tingkat internal ke tingkat fisik basis data. Detil *device-dependent* terhadap DBMS dan memberikan antar muka (*interface*) record tersimpan pada DBMS. Dengan demikian, maka antarmuka *record* tersimpan memungkinkan DBMS melihat struktur penyimpanan sebagai kumpulan dari file tersimpan.⁹

H. Fungsi

Fungsi merupakan relasi khusus. Kekhususan tersebut adalah : tidak ada pasangan yang satu anggota domain berpasangan dengan dua atau lebih anggota kodomain, ditulis : $f: A \rightarrow B$

⁹ Subekti, Muhammad. system manajemen basis data. ghalia Indonesia. 2004. hal 15-19

Definisi : Biarkan A dan B menjadi set. Fungsi dari A ke B adalah penugasan tepat satu elemen A. Kami menulis $f(x) = y$ jika y adalah elemen unik B yang ditugaskan oleh fungsi dari A ke B, kami menulis : $f: A \rightarrow B$ ¹⁰

Apabila anggota domain variable x dan anggota kodomain variable y, dan syarat tunggalnya y sudah terpenuhi maka definisi fungsi tersebut dapat dinyatakan:¹¹

$$f: A \rightarrow B = \{(x, y) | x \in A \text{ dan } y \in B, y = f(x)\}$$

Dengan demikian $f: A \rightarrow B$ dapat dituliskan $f(x) = y$ atau $y = f(x)$

Andaikan tiap unsur $x \in A$ berpadanan/berkaitan tepat satu unsur $y \in B$ maka kaitan itu membentuk suatu unsur fungsi A ke B. dalam aljabar biasanya fungsi dengan unsur x dari wilayah/domain A berkaitan dengan unsur y dari jangkau/range B dan ditulis: $f: A \rightarrow B$ dibaca f membawa x ke y dengan $x \in A$ dan $y \in B$ ¹²

Secara matematis di tuliskan

$$\forall x \in A \exists! y \in B \ni y: f(x)$$

Bentuk umum fungsi rasional adalah :

$$y = f(x) = \frac{P(x)}{Q(x)}$$

Dimana P dan Q adalah suku banyak. Daerah asal fungsi rasional adalah:

¹⁰ H.Rosen, Kenneth. Discrete Mathematics and its Applications. MCGRAW-Hill.

¹¹ Soedadyatmodjo dan wikaria gazali. Kalkulus. Graha ilmu. 2005. Hal 4

¹² A.B.Pangabea. Kalkulus. Graha ilmu. 2008. hal 7

$$D_f = \{x|Q(x) \neq 0\} = R - \{x|Q(x) = 0\}$$

Daerah hasil fungsi ini sukar ditentukan. Akan tetapi pada kasus P dan Q keduanya linear, daerah hasilnya dapat ditentukan dengan cara menyatakan x dalam y. demikian juga untuk kasus P dan Q linier atau kuadrat, daerah hasil fungsinya dapat ditentukan dengan memanfaatkan sifat diskriminasi dari bentuk kuadrat. Sebagai ilustrasi fungsi rasional adalah,

$$f(x) = \frac{x-1}{x^2+1}, g(x) = \frac{x^2+1}{x^2-1}$$

Bentuk umum fungsi irrasional adalah :

$$f(x) = \sqrt[n]{g(x)}, \text{ dimana } g \text{ fungsi rasional}$$

Daerah asal fungsi ini adalah :

$$D_f = \begin{cases} D_g & \text{, jika } n \text{ ganjil} \\ \{x|g(x) \geq 0\} & \text{, jika } n \text{ genap} \end{cases}$$

Daerah hasil fungsi ini sulit ditentukan. Dalam beberapa kasus, daerah hasil fungsi irrasional dapat ditentukan dengan melihat keistimewaan bentuknya dan berbagai sifat pertidaksamaan, atau memanfaatkan sifat lengkungan derajat dua. Sebagai ilustrasi fungsi irrasional yang cukup sederhana misalnya adalah :¹³

$$f(x) = \sqrt{4-x^2}, g(x) = \sqrt{9-(x-1)^2}$$

¹³ Prayudi. *Kalkulus Fungsi satu Variabel*. Graha ilmu. 2006.hal 39-40

I. Relasi menggunakan SQL

Standar bahasa SQL menggunakan kata tabel untuk menyatakan relasi, dan kebanyakan sering mengikuti aturan ini saat mendiskusikan tentang SQL. Subset dari SQL yang mendukung pembuatan, penghapusan, dan pengubahan tabel disebut data definition language (DDL). Selanjutnya, jika terdapat perintah yang memungkinkan pengguna mendefinisikan domain baru, seperti tipe perintah definisi dalam bahasa pemrograman.

Pernyataan *CREATE TABLE* digunakan untuk mendefinisikan sebuah tabel baru. Untuk membuat relasi student, dapat menggunakan pernyataan berikut:

```
CREATE TABLE Students (sid CHAR(20),
                        Name CHAR(20),
                        Login CHAR(20),
                        Age INTEGER,
                        Gpa REAL)
```

Tuple disisipkan dengan menggunakan perintah *INSERT*. Dapat juga menyisipkan tuple tunggal ke dalam tabel student sebagai berikut:

```
INSERT
INTO Student (sid,name,login,age,gpa)
VALUES      (53688,'smith',smith@ee',18,3.2)
```

Dapat juga menghilangkan daftar dari nama kolom dalam klausa *INTO* dan merinci nilainya pada urutan yang tepat, tetapi hal itu merupakan cara yang bagus untuk menjelaskan nama kolom.

Dapat juga menghapus tuple dengan menggunakan perintah *DELETE*. dapat menghapus semua tuple student dengan name setara dengan smith menggunakan perintah :

```
DELETE
FROM      Students S
WHERE     S.name = 'Smith'
```

Dapat juga mengubah nilai kolom dalam baris yang ada dengan menggunakan perintah *UPDATE*. Contohnya, kita dapat menambahkan age dan mengurangi gpa dari siswa dengan sid 53688;

```
UPDATE    Student S
SET        S.age=S.age + 1,S.gpa=S.gpa-1
WHERE     S.sid=53688
```

Contoh-contoh ini mengilustrasikan beberapa poin penting. Klausa *WHERE* digunakan terlebih dahulu dan menentukan baris mana yang akan diubah. Klausa *SET* kemudian menentukan bagaimana cara baris itu diubah. Jika kolom yang diubah juga digunakan untuk menentukan nilai baru, maka nilai yang digunakan dalam ekspresi pada sisi kanan tanda sama dengan (=) adalah nilai yang lama, yaitu sebelum

pengubahan.untuk mengilustrasikan poin-poin ini lebih lanjut, pikirkan beberapa variasi dari query sebelumnya, seperti di bawah ini:

```
UPDATE      Student S
SET         S.gpa=S.gpa-0.1
WHERE      S.gpa >= 3.3
```

Jika *query* ini diaplikasikan pada contoh S1 pada *Students* seperti yang digunakan pada dambar 2.3, kita akan mendapatkan contoh seperti yang ditunjukkan pada gambar 2.4.¹⁴

Tabel 2.3 *Students* S1 sebelum pembaruan

<i>sid</i>	<i>name</i>	<i>Login</i>	<i>age</i>	<i>gpa</i>
50000	Dave	<u>dave@cs</u>	19	3.3
53666	Jones	<u>jones@cs</u>	18	3.4
53688	Smith	<u>smith@cs</u>	18	3.2
53650	Smith	<u>smith@math</u>	19	3.8
53831	Madayan	<u>madayan@music</u>	11	1.8
53832	Guldu	<u>guldu@music</u>	12	2.0

Tabel 2.4 *Students* S1 setelah pembaruan

¹⁴ Johannes gehrke,reghu ramakrishnan, *Sistem Manajemen Database*, Andi and McGraw-Hill Education,2004.. hal 46-47

<i>sid</i>	<i>name</i>	<i>login</i>	<i>age</i>	<i>gpa</i>
50000	Dave	<u>dave@cs</u>	19	3.2
53666	Jones	<u>jones@cs</u>	18	3.3
53688	Smith	<u>smith@cs</u>	18	3.2
53650	Smith	<u>smith@math</u>	19	3.7
53831	Madayan	<u>madayan@music</u>	11	1.8
53832	Guldu	<u>guldu@music</u>	12	2.0

J. Bahasa SQL

Structured Query Language (SQL) adalah bahasa *database* relasional yang penggunaannya paling luas. Bahasa ini pertama kali dikembangkan di IBM melalui proyek *SEQUEL-XRM* dan System-R (1974-1977). Tak lama kemudian, vendor lain memperkenalkan produk DBMS yang berbasis SQL, dan hal itu sekarang menjadi standar *defacto*. SQL terus berkembang sebagai respons terhadap perubahan kebutuhan dalam bidang database. Standar ANSI/ISO untuk SQL saat ini disebut SQL:1999. Meski belum seluruh produk DBMS mendukung standar SQL:1999 secara penuh, vendor terus berusaha mewujudkan dan sebagian besar produk telah mendukung fitur-fitur intinya. Standar SQL:1999 tidak jauh berbeda dengan standar sebelumnya, yaitu SQL-92, dengan mengacu pada fitur yang akan kami bahas dalam bab ini. Presentasi kita konsisten dengan SQL-92 dan SQL:1999, dan kita secara eksplisit membahas beberapa aspek yang membedakan kedua versi standar tersebut.

Bahasa SQL memiliki beberapa aspek pengenalan:

1. Data Manipulasi *language* (DML): Subset SQL ini memungkinkan pengguna untuk mengajukan *query* dan untuk menyisipkan, menghapus, dan memodifikasi baris.
2. Data *Definition Language* (DDL): Subset SQL ini mendukung pembuatan, penghapusan dan modifikasi definisi untuk tabel dan view. Batasan integritas dapat dinyatakan dalam tabel, baik pada saat tabel dibuat maupun sesudah
3. *Trigger* dan batasan integritas lanjut : Standar SQL:1999 baru mencakup dukungan untuk *trigger*, yaitu tindakan yang dilakukan oleh DBMS ketika perubahan pada database sesuai dengan syarat yang ditentukan pada *trigger*. Kita membahas *trigger* dalam bab ini. SQL memungkinkan penggunaan *Query* untuk menyatakan spesifikasi batasan integritas kompleks.
4. *Embedded SQL* dan *dynamic SQL* : Fitur *SQL embedded* memungkinkan kode SQL dipanggil dari host language seperti C. atau COBOL. Fitur *SQL dynamic* memungkinkan *query* dibangun dan dijalankan pada saat runtime
5. *Client-server Execution* dan *remote database acces* : perintah ini mengatur bagaimana program aplikasi client dihubungkan ke server database SQL, atau mengakses data dari database melalui jaringan
6. Manajemen Transaksi : Beragam perintah yang memungkinkan pengguna untuk mengontrol aspek yang berkaitan dengan jalannya transaksi secara eksplisit
7. Keamanan (*security*): SQL menyediakan mekanisme untuk mengontrol akses pengguna ke obyek data seperti tabel dan gambar.

8. Fitur-fitur lanjutan : Standar SQL:1999 mencakup fitur yang berorientasi pada obyek, *Query* rekursil, *Query* pendukung keputusan dan juga bidang baru seperti data mining, data special serta manajemen teks dan data XML.

Sintaksis query SQL dasar yaitu:

- a) From-list pada klausa *FROM* adalah daftar nama tabel. Nama tabel dapat diikuti oleh variable rentang; variable rentang khususnya berguna ketika nama tabel yang sama muncul lebih dari sekali pada form list.
- b) *Select-list* adalah daftar nama kolom (termasuk ekspresinya) dari tabel-tabel yang tercantum pada *form list*. Nama kolom dapat diawali dengan variable rentang.
- c) Kualifikasi pada klausa *WHERE* merupakan kombinasi boolean (yaitu pernyataan dengan kata sambung logika *AND*, *OR*, dan *NOT*) dari kondisi yang berbentuk ekspresi op ekspresi, dimana op adalah salah satu dari operator pembandingan {<, <=, =, <>, >=, >}. Ekspresi merupakan nama kolom, sebuah konstanta, atau ekspresi (aritmatika atau string)
- d) Kata kunci *Distinct* bersifat pilihan. Hal itu menunjukkan bahwa tabel yang dihitung sebagai jawaban untuk query ini tidak boleh memiliki duplikat, yaitu memiliki dua salinan pada baris yang sama. Default-nya adalah bahwa duplikat itu tidak dihilangkan¹⁵

¹⁵ Gehrke, Johannes dan Ramakrishnan. *Sistem Manajemen Database*. Andi and McGraw-Hill Education. 2004. hal 102

K. Query sebagai ekspresi aljabar relasional

Langkah pertama mengoptimalkan blok *query* mengekspresikan sabagai ekspresi aljabar relasional. Untuk keseragaman, kita asumsikan bahwa *group by* dan *having* juga oppertor dalam aljabar yang diperluas, yang digunakan untuk rencana, dan bahwa operasi agregasi bias muncul dalam daftar argument operator proyeksi. Arti operator tentunya telah jelas dari pembahasan SQL.¹⁶

SQL query pada gambar :

```
SELECT S.sid, MIN(R.day)
FROM Sailors S, Reserves R, Boats B
WHERE S.sid=R.sid AND R.bid AND B.color='red' AND
      S.rating=Reference to nested block
GROUP BY S.sid
HAVING COUNT (*) > 1
```

Dapat diekspresikan dalam aljabar yang diperluas sebagai berikut :

```
 $\Pi_{S.sid, MIN(R>day)} ($ 
 $HAVING_{COUNT(*) > 2} ($ 
 $GROUP BY_{S.sid} ($ 
 $\sigma_{S.sid=R.sid \wedge B.color='red' \wedge S.rating=value\_from\_nested\_block} ($ 
 $Sailors \times reserves \times Boats))$ 
```

¹⁶ Gehrke ,Johannes dan reghu ramakrishnan. *Sistem Manajemen Database*. Andi and McGraw-Hill Education.2004.hal 365

BAB III

METODE PENELITIAN

A. Jenis penelitian

Adapun jenis penelitian yang digunakan adalah penelitian terapan

B. Waktu Penelitian

Adapun Waktu penelitian dari Januari sampai Maret 2019

C. Prosedur penelitian

Setelah melakukan pengambilan data, langkah yang di gunakan dalam menyelesaikan penelitian ini adalah:

1. Menentukan tabel yang ingin di kelolah dari *database* ruang baca.
2. Menentukan data tabel pada *database* yang ingin diolah.
3. Mengolah database menggunakan operator-operator aljabar relasional.
 1. Data *manipulation Language* (DML), dimana perintah menambahkan tabel, *projection*, *union*, *insert*, *delete* dan *update*.
 2. Data *Definition Language* (DDL) dimana perintah *rename*, *drop* dan menambahkan tupel

BAB IV

HASIL DAN PEMBAHASAN

A. Hasil Penelitian

Adapun hasil penelitian dari data yang diambil dari Ruang baca jurusan Matematika Fakultas Sains dan teknologi.

1. Tabel yang di gunakan dalam *database* ruang baca.

1. Struktur *Database* Biblio

Struktur database yang di gunakan dalam pengujian *database* yang dimana diinput pada aplikasi slims, yang terhubung dengan php.

Tabel 4.1 Struktur *Database* Biblio

nama	type	width	condition
biblio_id	int	11	not null
gmd_id	int	3	null
title	text		not null
sor	varchar	200	null
edition	varchar	50	null
isbn_issn	vrchar	20	null
publisher_id	int	11	null
Publish_year	varchar	20	null
collation	varchar	50	null
series_title	varchar	200	null
call_number	varchar	50	null
language_id	char	5	null
source	varchar	3	null
publish_place_id	int	11	null
Classification	varchar	40	null
Notes	text		null
Image	varchar	100	null
File_att	varchar	255	null
Opac_hide	smallint	1	null
Promoted	smallint	1	null
Labes	text		null
Frecuency_id	int	11	not null
Spec_Detail_info	text		null
content_type_id	int	11	null
media_type_id	int	11	null
Carrier_type_id	int	11	null
Input_date	datetime		null
last update	datetime		null

Tabel 4.1 Merupakan struktur *database* pada Tupel Biblio yang dimana salah satu tabel yang di gunakan dalam pengolahan *database* .

2. Struktur Database Biblio_Author

Struktur *database* yang di gunakan dalam pengujian database pada aplikasi slims, yang terhubung dengan php.

Tabel 4.2 Struktur Database Biblio_Author

Name	type	width	condition
Biblio_id	int	11	not null
Author_id	int	11	not null
Level	int	1	not null

Tabel 4.2 Merupakan struktur *database* pada tabel biblio_author yang dimana salah satu tabel yang di gunakan dalam pengolahan *database* .

2. Data Manipulation Language (DML)

DML merupakan kumpulan perintah SQL yang dimana menampilkan data, *projection*, *union*, *insert*, *delete*, dan *update*.

1. Menampilkan Data

Operasi Menampilkan Data berfungsi untuk menyeleksi tuple dalam suatu relasi yang memenuhi predikat yang diberikan dari sebuah tabel relasi. Simbol sigma (σ) digunakan untuk menunjukkan operasi Menampilkan Data.

Tabel 4.3 Table biblio

biblio_id	gmd_id	title	sor	edition	isbn_isbn	Publisher_id	call_number
1	1	analisis deret dan waktu				0	519.4
2	1	statistika untuk penelitian				0	519.4
3	1	pengantar teori bilangan					
4	1	teori grub					
5	1	teknik penarikan sampel					

Tabel 4.3 adalah tabel biblio yang di gunakan dalam menampilkan table secara keseluruhan ataupun berdasarkan syarat tertentu

Dalam menampilkan data secara menyeluruh dapat di tuliskan dalam bentuk.

Aljabar Relasional :

σ = biblio

Berdasarkan Aljabar Relasional menampilkan data secara menyeluruh dapat di tuliskan dalam Sintaks SQL :

SELECT * FROM `biblio`

Berdasarkan sintaks SQL dapat di jelaskan bahwa *select* adalah perintah yang di gunakan untuk menampilkan nilai atau recor yang di pilih, tanda * digunakan untuk memilih semua isi tabel, from perintah dari yang mendeklarasikan ke tabel yang ingin di kelolah, Biblio sebagai nama tabel. Pada perintah ini tidak memiliki syarat maka semua data pada tabel biblio akan di tampilkan.

Tabel 4.4 Hasil dari menampilkan data tanpa syarat

biblio_id	gmd_id	title	sor	edition	isbn_isbn	Publisher_id	call_number
1	1	analisis deret dan waktu				0	519.4
2	1	statistika untuk penelitian				0	519.4
3	1	pengantar teori bilangan					
4	1	teori grub					
5	1	teknik penarikan sampel					

Tabel 4.4 pada pengujian menampilkan data secara menyeluruh atau tidak menggunakan syarat tertentu

Berdasarkan perintah menampilkan data tanpa menggunakan perintah bersyarat yang tertera pada tabel 4.4 merupakan hasil dari tabel 4.3, yang dimana kedua tabel tidak memiliki perbedaan yang mengartikan bahwa tabel yang tertera pada tabel 4.4 adalah tabel yang menampilkan secara keseluruhan pada tabel 4.3

Dalam menampilkan data secara bersyarat dapat di tuliskan dalam bentuk.

Aljabar Relasional :

$\sigma_{\text{call_number} = '519.4'}(\text{biblio})$

Berdasarkan Aljabar Relasional menampilkan data bersyarat dapat di tuliskan dalam Sintaks SQL :

```
SELECT * FROM `biblio` WHERE `call_number`='519.4'
```

Berdasarkan sintaks SQL dapat di jelaskan bahwa select adalah perintah yang di gunakan untuk menampilkan nilai atau *recor* yang di pilih, tanda * digunakan untuk memilih semua isi tabel, *from* perintah dari yang mendeklarasikan ke tabel yang ingin di kelolah, Biblio sebagai nama tabel, *where* sebagai perintah syarat, dan *call_number = 519* adalah kriteria atau kondisi. Pada perintah ini memiliki syarat maka data yang di tampilkan yang memenuhi syarat.

Tabel 4.5 Hasil dari menampilkan data bersyarat

biblio_id	gmd_id	title	sor	edition	isbn_isbn	Publisher_id	call_number
1	1	analisis deret dan waktu				0	519.4
2	1	statistika untuk penelitian				0	519.4

Tabel 4.5 pada pengujian menampilkan data menggunakan syarat, menampilkan *call_number* 5.19

Dalam menampilkan data yang perlu di perhatikan pada tabel 4.5 menghasilkan data yang tidak sama yang tertera pada tabel 4.3 yang mengartikan data yang di tampilkan secara bersyarat. Ini mengartikan tidak semua tabel akan di tampilkan pada tabel 4.5, yang dimana yang di tampilkan hanya tabel yang memenuhi syarat $\text{call_number} = 519.4$.

2. *Projection*

Operasi projection berfungsi untuk memperoleh kolom-kolom tertentu dari sebuah tabel relasi. Simbol phi (π) digunakan untuk menunjukkan operasi projection.

Tabel 4.6 Tabel biblio pada pengujian Projection

biblio_id	gmd_id	title	sor	edition	isbn_isbn	Publisher_id	call_number
1	1	analisis deret dan waktu				0	519.4
2	1	statistika untuk penelitian				0	519.4
3	1	pengantar teori bilangan					
4	1	teori grub					
5	1	teknik penarikan sampel					

Tabel 4.6 adalah tabel biblio yang di gunakan dalam menguji penggunaan Projection menggunakan manipulasi Aljabar Relasional

Dalam perintah *Projection* secara menyeluruh dapat di tuliskan dalam bentuk.

Aljabar Relasional :

$\pi_{\text{biblio_id}, \text{title}, \text{call_number}} (\text{biblio})$

Berdasarkan Aljabar Relasional Projection secara menyeluruh dapat di tuliskan dalam Sintaks SQL :

```
SELECT `biblio_id`,`title`,`call_number` FROM `biblio`
```

Berdasarkan sintaks SQL dapat di jelaskan bahwa *select* adalah perintah yang di gunakan untuk menampilkan nilai atau *recor* yang di pilih , Biblio_id, title dan call_number sebagai nama kolom, *from* perintah dari, dan biblio sebagai nama tabel.

Tabel 4.7 Hasil Pengujian Projection tanpa bersyarat

biblio_id	title	call_number
1	analisis deret dan waktu	519.4
2	statistika untuk penelitian	519.4
3	pengantar teori bilangan	
4	teori grub	
5	teknik penarikan sampel	

Tabel 4.7 pada pengujian Projection menampilkan kolom pada tabel, dimana menampilkan tabel biblio_id, title dan call_number pada tabel Biblio tanpa menggunakan perintah *where* atau bersyarat.

Dalam perintah projection yang perlu di perhatikan pada tabel 4.6 nama kolom yang ingin di tampilkan sehingga yang tertera pada tabel 4.7 sesuai dengan yang ingin di tampilkan, pada tabel 4.7 menampilkan tabel secara keseluruhan sesuai dengan perintah SQL tanpa menggunakan perintah bersyarat atau *where*. Dimana pada hasil pengujian *projection* tanpa perintah bersyarat tertera pada tabel 4.7 yang dimana pengujian dari tabel 4.6.

Dalam perintah *Projection* secara bersyarat dapat di tuliskan dalam bentuk.

Aljabar Relasional :

```
Πcall_number = `519.4` biblio_id, title, call_number
(biblio)
```

Berdasarkan Aljabar Relasional Projection bersyarat dapat di tuliskan dalam

Sintaks SQL :

```
SELECT `biblio_id`,`title`,`call_number` FROM `biblio`
WHERE `call_number`=519.4
```

Berdasarkan sintaks SQL dapat di jelaskan bahwa select adalah perintah yang di gunakan untuk menampilkan nilai atau recor yang, Biblio_id, title dan call_number sebagai nama kolom, *from* perintah dari, *where* perintah bersyarat, dan biblio sebagai nama tabel, call_number sebagai kondisi. dimana dalam pengolahan ini menampilkan kolom yang di tampilkan berdasarkan syarat call_number = 519.4

Tabel 4.8 Hasil Pengujian Projection bersyarat

biblio_id	title	call_number
1	analisis deret dan waktu	519.4
2	statistika untuk penelitian	519.4

Tabel 4.8 pada pengujian Projection menampilkan kolom pada tabel, dimana menampilkan tabel biblio_id, title, call_number pada tabel Biblio dengan menggunakan perintah bersyarat yaitu call_number =519.4.

Dalam perintah *projection* yang perlu di perhatikan pada tabel 4.6 nama kolom yang ingin di tampilkan sehingga yang muncul pada tabel 4.8 sesuai

dengan yang ingin di tampilkan, pada tabel 4.8 menampilkan kolom pada tabel secara bersyarat, dimana kolom yang di tampilkan yang memenuhi syarat *where*, sehingga hanya kolom tertentu saja yang tampil. Dimana perintah bersyarat yaitu `call_number = 519.4`

3. *Union*

Operasi *union* berfungsi untuk mendapatkan gabungan nilai atribut dari sebuah tabel relasi dengan nilai atribut dari tabel relasi lainnya. Simbol (\cup) digunakan untuk menunjukkan operasi union.

Tabel 4.9 Table Biblio Pada pengujian Union

bibli_id	gmd_id	title
120	1	Praktikum Microsoft Windows Vista
121	1	Minitab Programming
122	1	Keterampilan Dasar Pengoperasian Komputer
123	1	123 Jurus Mengoptimalkan Windows Vista Menjadi Mah
124	1	Differential Equations DeMyStiFieD

Tabel 4.9 merupakan tabel biblio yang di gunakan dalam menguji penggunaan Union menggunakan manipulasi Aljabar Relasional dimana jumlah baris pada tabel secara keseluruhan 124

Tabel 4.10 Table Biblio_author Pada pengujian Union

biblio_id	author_id	level
120	97	1
121	98	1
122	99	1
123	100	1
124	101	1

Tabel 4.10 adalah tabel biblio_author yang di gunakan dalam menguji penggunaan Union menggunakan manipulasi Aljabar Relasional dimana jumlah baris pada tabel secara keseluruhan 124

Dalam percobaan Union yaitu menggabungkan dua tabel yaitu tabel biblio dan biblio_author.

Aljabar Relasional :

$\Pi_{\text{biblio_id}}(\text{biblio}) \cup \text{biblio_id}(\text{biblio_author})$

Berdasarkan Aljabar Relasional secara bersyarat dapat di tuliskan dalam

Sintaks SQL :

```
SELECT * FROM `biblio` AS A, `biblio_author` AS B WHERE B.`biblio_id`=A.`biblio_id`
```

Berdasarkan sintaks SQL dapat di jelaskan bahwa select adalah perintah yang di gunakan untuk menampilkan nilai atau *recor* yang, tanda * digunakan untuk memilih semua isi tabel, *from* merupakan perintah dari, Biblio dan Biblio_author sebagai nama tabel, AS A dan AS B diartikan sebagai tabel A dan tabel B yang ingin di gabungkan, *where* sebagai perintah bersyarat, dan B.`biblio_id`=A.`biblio_id` sebagai kondisi atau syarat.

Tabel 4.11 Hasil Pengujian *Union* Bersyarat

bibli_id	gmd_id	title	biblio_id	author_id	level
120	1	Praktikum Microsoft Windows Vista	120	97	1
121	1	Minitab Programming	121	98	1
122	1	Keterampilan Dasar Pengoperasian Komputer	122	99	1
123	1	123 Jurusan Mengoptimalkan Windows Vista Menjadi Mah	123	100	1
124	1	Differential Equations DeMyStiFieD	124	101	1

Tabel 4.11 pada pengujian *Union* yaitu menggabung dua tabel, yaitu tabel biblio dan biblio_author dengan syarat berdasarkan Biblio_id.

Dimana pada hasil pengujian *Union* bersyarat tertera pada tabel 4.11 yang dimana penggabungan tabel 4.9 dan tabel 4.10, Adapun tabel yang digabung adalah tabel biblio dan biblio_author. Dimana data digabungkan berdasarkan berdasarkan kolom biblio_id.

4. *Insert*

Insert Digunakan untuk menambahkan data pada suatu tabel atau relasi .

Simbol (U) digunakan untuk menunjukkan operasi *insert*.

Table 4.12 Table Biblio pada pengujian insert

bibli_id	gmd_id	title
120	1	Praktikum Microsoft Windows Vista
121	1	Minitab Programming
122	1	Keterampilan Dasar Pengoperasian Komputer
123	1	123 Jurusan Mengoptimalkan Windows Vista Menjadi Mah
124	1	Differential Equations DeMyStiFieD

Tabel 4.12 adalah tabel biblio yang di gunakan dalam menguji penggunaan *Insert* menggunakan manipulasi Aljabar Relasional

Pada perintah *Insert* yang di lakukan adalah menambah data pada tabel biblio.

Aljabar Relasional :

Biblio U { (biblio_id,gmd_id,title) = (125,1,teori
bilangan) }

Berdasarkan Aljabar Relasional Perintah insert dapat di tuliskan dalam Sintaks SQL :

```
INSERT INTO `biblio`(`biblio_id`, `gmd_id`, `title`)
VALUES (125,1,teori bilangan)
```

Berdasarkan sintaks SQL dapat di jelaskan bahwa insert into perintah untuk memperbarui data, biblio sebagai nama tabel, Biblio_id, gmd_id dan title sebagai nama kolom yang ingin ditambahkan, *values* perintah yang mendeklarasikan nilai atau data baru dan nilai 125, 1, dan teori bilangan sebagai tupel yang dimasukkan dalam suatu tabel.

Tabel 4.13 Hasil Pengujian *Insert*

bibli_id	gmd_id	title
120	1	Praktikum Microsoft Windows Vista
121	1	Minitab Programming
122	1	Keterampilan Dasar Pengoperasian Komputer
123	1	123 Jurus Mengoptimalkan Windows Vista Menjadi Mah
124	1	Differential Equations DeMyStiFieD
125	1	Teori bilangan

Tabel 4.13 pada pengujian insert yaitu menambah data baru dimana data yang di tambahkan biblio_id = 125, gmd_id = 1, title = teori bilangan pada tabel biblio.

Dalam menambahkan nilai baru pada tabel 4.12 yang perlu di perhatikan adalah jumlah tabel dan urutan tabel sehingga nilai yang di input pada tabel 4.13 sesuai dengan yang di inginkan.

5. *Delete*

Delete Digunakan untuk menghapus suatu data tertentu pada suatu tabel.

Simbol (-) digunakan untuk menunjukkan operasi *Delete*.

Table 4.14 Table biblio Pada pengujian *Delete*

Biblio_id	gmd_id	title
1	1	Analisis deret dan waktu
2	1	Statistika Untuk Penelitian
3	1	Pengantar Teori Bilangan
4	1	Teori Grup
5	1	Teknik Penarikan Sampel
6	1	Kalkulus dan Geometri analisis jilid 2
7	1	Aljabar rendah jilid III
8	1	Geometri Analitik Bidang Dan Ruang
9	1	Pengendalian Kualitas Statistika
10	1	Belajar Pemrograman VB6 Dalam Sekejap
11	1	Statistika Dan Rancang Percobaan
12	1	Pengendalian Kualitas Statistika
13	1	Regresi Dan Korelasi Dalam Penggandaan
14	1	Statistika Deskriptif dalam Bidang ekonomi dan Niaga
15	1	Teknik statistika untuk Bisnis & ekonomi Edisi kesembilan jilid 1
16	1	Teknik statistika untuk Bisnis & ekonomi Edisi kesembilan jilid 2
17	1	Teknik statistika untuk Bisnis & ekonomi Edisi kesembilan jilid 3

Tabel 4.14 adalah tabel biblio yang di gunakan dalam menguji penggunaan *Delete* menggunakan manipulasi Aljabar Relasional.

Pada perintah *Delete* yaitu menghapus tabel biblio secara keseluruhan.

Aljabar Relasional :

Biblio - σ Biblio

Berdasarkan Aljabar Relasional *Delete* secara keseluruhan dapat di tuliskan dalam Sintaks SQL :

```
DELETE FROM `biblio`
```

Berdasarkan sintaks SQL dapat di jelaskan bahwa *Delete* dapat di artikan Menghapus *record* atau nilai didalam suatu kolom pada tabel tertentu, *from* merupakan perintah dari, Biblio sebagai nama tabel yang artinya semua pada tabel biblio dihapus.

Tabel 4.15 Hasil Pengujian *Delete* tanpa bersyarat

Biblio_id	gmd_id	title

Table 4.15 Pada Pengujian *Delete* yaitu menghapus data Biblio secara keseluruhan

Pada tabel 4.15 adalah hasil dari *delete* pada tabel 4.14. dimana menghapus isi tabel secara menyeluruh. Pada perintah *delete* hanya menghapus isi data pada tabel namun tidak mempengaruhi struktur pada tabel. Untuk membuktikan ulang apakah isi dari tabel biblio sudah terhapus, dengan cara perintah `SELECT * FROM 'biblio'`

Dalam percobaan Operasi *Delete* yaitu menghapus tabel biblio secara bersyarat atau tidak secara menyeluruh.

Aljabar Relasional :

$\text{Biblio} - \sigma_{\text{biblio_id}=1}$

Berdasarkan Aljabar Relasional *Delete* secara bersyarat dapat di tuliskan dalam Sintaks SQL :

```
DELETE FROM `biblio` WHERE `biblio_id`=1
```

Tabel 4.16 Hasil Pengujian *Delete* bersyarat

Biblio_id	gmd_id	Title
2	1	Statistika Untuk Penelitian
3	1	Pengantar Teori Bilangan
4	1	Teori Grup
5	1	Teknik Penarikan Sampel
6	1	Kalkulus Dan Geometri Analisis Jilid 2
7	1	Aljabar Rendah Jilid III
8	1	Geometri Analitik Bidang Dan Ruang
9	1	Pengendalian Kualitas Statistika
10	1	Belajar Pemrograman VB6 Dalam Sekejap
11	1	Statistika Dan Rancang Percobaan
12	1	Pengendalian Kualitas Statistika
13	1	Regresi Dan Korelasi Dalam Penggunaan
14	1	Statistika Deskriptif dalam Bidang ekonomi dan Niaga
15	1	Teknik statistika untuk Bisnis & ekonomi Edisi kesembilan jilid 1
16	1	Teknik statistika untuk Bisnis & ekonomi Edisi kesembilan jilid 2
17	1	Teknik statistika untuk Bisnis & ekonomi Edisi kesembilan jilid 3

Table 4.16 Pada Pengujian *Delete* bersyarat yaitu menghapus data dimana biblio_id = 1 pada tabel biblio

Pada tabel 4.16 adalah hasil dari *delete* pada tabel 4.14. dimana menghapus isi tabel secara bersyarat dimana biblio_id = 1 . Pada perintah *delete* hanya menghapus isi data pada tabel namun tidak mempengaruhi struktur pada tabel. Untuk membuktikan ulang apakah buku analisis deret dan waktu sudah terhapus dari tabel biblio `SELECT * FROM `biblio` where biblio_id =1`

6. Update

Update adalah memperbaharui data lama ke dalam bentuk yang lebih baru. Simbol (δ) digunakan untuk menunjukkan operasi *Update*.

Table 4.17 Table Biblio Pada pengujian update

Biblio_id	gmd_id	title
1	1	Analisis deret dan waktu
2	1	Statistika Untuk Penelitian
3	1	Pengantar Teori Bilangan
4	1	Teori Grup
5	1	Teknik Penarikan Sampel
6	1	Kalkulus dan Geometri analisis jilid 2
7	1	Aljabar rendah jilid III
8	1	Geometri Analitik Bidang Dan Ruang
9	1	Pengendalian Kualitas Statistika
10	1	Belajar Pemrograman VB6 Dalam Sekejap
11	1	Statistika Dan Rancang Percobaan
12	1	Pengendalian Kualitas Statistika
13	1	Regresi Dan Korelasi Dalam Penggandaan
14	1	Statistika Deskriptif dalam Bidang ekonomi dan Niaga
15	1	Teknik statistika untuk Bisnis & ekonomi Edisi kesembilan jilid 1
16	1	Teknik statistika untuk Bisnis & ekonomi Edisi kesembilan jilid 2
17	1	Teknik statistika untuk Bisnis & ekonomi Edisi kesembilan jilid 3

Tabel 4.17 adalah tabel biblio yang di gunakan dalam menguji penggunaan *update* menggunakan manipulasi Aljabar Relasional

Dalam percobaan Operasi *Update* yaitu memperbaharui kolom title secara keseluruhan.

Aljabar Relasional :

$title = \text{Analisi Deret}(\delta \text{biblio})$

Berdasarkan Aljabar Relasional *Update* secara keseluruhan dapat di tuliskan dalam Sintaks SQL :

```
UPDATE `biblio` SET `title`="Analisis Deret"
```

Berdasarkan sintaks SQL dapat di jelaskan bahwa *update* Berfungsi untuk merubah *record* atau nilai didalam suatu kolom, Biblio sebagai nama tabel, set sebagai perintah ke nama kolom, title sebagai nama kolom, maka

semua yang ada pada tabel biblio kolom title berubah menjadi analisis deret dan waktu .

Tabel 4.18 Pengujian Update secara keseluruhan

Biblio_id	gmd_id	title
1	1	Analisis deret dan waktu
2	1	Analisis deret dan waktu
3	1	Analisis deret dan waktu
4	1	Analisis deret dan waktu
5	1	Analisis deret dan waktu
6	1	Analisis deret dan waktu
7	1	Analisis deret dan waktu
8	1	Analisis deret dan waktu
9	1	Analisis deret dan waktu
10	1	Analisis deret dan waktu
11	1	Analisis deret dan waktu
12	1	Analisis deret dan waktu
13	1	Analisis deret dan waktu
14	1	Analisis deret dan waktu
15	1	Analisis deret dan waktu
16	1	Analisis deret dan waktu
17	1	Analisis deret dan waktu

Tabel 4.18 Pada pengujian *Update* tanpa bersyarat atau secara keseluruhan yaitu merubah semua yang ada pada kolom title menjadi analisis deret dan waktu

Dalam memperbaharui tabel 4.17 sehingga menjadi tabel 4.18 dengan memperbaharui semua title pada tabel biblio menjadi analisis deret dan waktu, yang di mana memperbaharui secara keseluruhan pada tabel.

Dalam percobaan Operasi *Update* yaitu memperbaharui kolom title secara bersyarat.

Aljabar Relasional :

```
title = Analisi Deret (  $\delta$ biblio_id = 1) (biblio)
```

Berdasarkan Aljabar Relasional *Update* secara bersyarat dapat di tuliskan dalam Sintaks SQL :

```
UPDATE `biblio` SET `title`="Analisis Deret" WHERE
`biblio_id`=1
```

Berdasarkan sintaks SQL dapat di jelaskan bahwa update Berfungsi untuk merubah *record* atau nilai didalam suatu kolom, Biblio sebagai nama tabel, set sebagai perintah ke nama kolom, title sebagai nama kolom, analisis deret sebagai nama baru yang ingin di pakai dalam pembaharuan, *where* perintah bersyarat, *biblio_id* =1 merupakan syarat atau kondisi.

Tabel 4.19 Pengujian *Update* Bersyarat

Biblio_id	gmd_id	Title
1	1	Analisis deret
2	1	Statistika Untuk Penelitian
3	1	Pengantar Teori Bilangan
4	1	Teori Grup
5	1	Teknik Penarikan Sampel
6	1	Kalkulus Dan Geometri Analisis Jilid 2
7	1	Aljabar rendah jilid III
8	1	Geometri Analitik Bidang Dan Ruang
9	1	Pengendalian Kualitas Satatistika
10	1	Belajar Pemrograman VB6 Dalam Sekejap
11	1	Statistika Dan Rancang Percobaan
12	1	Pengendalian Kualitas Satatistika
13	1	Regresi Dan Korelasi Dalam Penggandaan
14	1	Statistika Deskriptif dalam Bidang ekonomi dan Niaga
15	1	Teknik statistika untuk Bisnis & ekonomi Edisi kesembilan jilid 1
16	1	Teknik statistika untuk Bisnis & ekonomi Edisi kesembilan jilid 2
17	1	Teknik statistika untuk Bisnis & ekonomi Edisi kesembilan jilid 3

Tabel 4.19 Pada pengujian *Update* yaitu merubah nama analisis deret dan waktu menjadi analisis deret

Dimana yang tertera pada tabel 4.19 merupakan hasil dari pembaruan dengan perintah *update* menggunakan perintah bersyarat pada tabel 4.17.

dimana yang di perbaharui atau di *update* hanya yang memenuhi syarat `biblio_id = 1`.

3. Data Definiton Language (DDL)

DDL merupakan kumpulan perintah SQL yang dimana, rename, drop dan menambahkan tabel.

1. Rename

Rename Digunakan untuk merubah nama lama menjadi nama yang baru. Simbol (ρ) digunakan untuk menunjukkan operasi *Rename*.

Table 4.20 Jenis table dalam *database* Slims

table
backup_log
biblio
biblio_attachment
biblio_author
biblio_custom

Tabel 4.20 adalah jenis tabel dalam *database* slim yang di gunakan dalam menguji

Dalam perintah *Rename* mengubah nama tupel `biblio` menjadi `nama_buku` pada *database*.

Aljabar Relasional :

ρ nama_buku^(biblio)

Berdasarkan Aljabar Relasional perintah *rename* dapat di tuliskan dalam

Sintaks SQL :

```
ALTER TABLE biblio RENAME TO nama_buku
```

Berdasarkan sintaks SQL dapat di jelaskan bahwa alter table digunakan untuk merubah atribut pada suatu tabel, Biblio sebagai nama tabel, *rename* perintah mengubah nama, nama_buku sebagai nama baru perubahan dari biblio.

Tabel 4.21 Tabel pengujian *Rename*

table
backup_log
nama_buku
biblio_attachment
biblio_author
biblio_custom

Tabel 4.21 pada pengujian *rename* yaitu mengubah nama tabel dari biblio ke nama_buku

Dalam merubah nama tupel pada tabel 4.20 yaitu biblio menjadi nama_buku yang tertera pada tabel 4.21. yang perlu diketahui bahwa perubahan nama tupel dapat mempengaruhi struktur pada *database*.

Dalam perintah *rename* mengubah nama kolom pada tabel title menjadi judul_buku

Table 4.22 Table biblio Pada pengujian *Rename*

Biblio_id	gmd_id	title
1	1	Analisis deret dan waktu
2	1	Statistika Untuk Penelitian
3	1	Pengantar Teori Bilangan
4	1	Teori Grup
5	1	Teknik Penarikan Sampel
6	1	Kalkulus dan Geometri analisis jilid 2
7	1	Aljabar rendah jilid III
8	1	Geometri Analitik Bidang Dan Ruang
9	1	Pengendalian Kualitas Satatistika
10	1	Belajar Penrograman VB6 Dalam Sekejap
11	1	Statistika Dan Rancang Percobaan
12	1	Pengendalian Kualitas Satatistika
13	1	Regresi Dan Korelasi Dalam Penggandaan
14	1	Statistika Deskriptif dalam Bidang ekonomi dan Niaga
15	1	Teknik statistika untuk Bisnis & ekonomi Edisi kesembilan jilid 1
16	1	Teknik statistika untuk Bisnis & ekonomi Edisi kesembilan jilid 2
17	1	Teknik statistika untuk Bisnis & ekonomi Edisi kesembilan jilid 3

Tabel 4.22 adalah tabel biblio yang di gunakan dalam menguji penggunaan *Rename* menggunakan manipulasi Aljabar Relasional

Aljabar Relasional :

ρ (nama_buku) (judul_buku) ^(title)

Dalam percobaan Operasi *Rename* menggunakan manipulasi Aljabar Relasional.

SQL :

```
ALTER TABLE biblio CHANGE title judul_buku Char(15);
```

Berdasarkan sintaks SQL dapat di jelaskan bahwa alter table digunakan untuk merubah atribut pada suatu tabel, Biblio sebagai nama tabel, judul_buku sebagai nama kolom baru dari pembaharuan title. Char(15) adalah jumlah maksimal huruf atau penulisan pada judul.

Tabel 4.23 Hasil pengujian *Rename*

Biblio_id	gmd_id	Judul_buku
1	1	Analisis deret dan waktu
2	1	Statistika Untuk Penelitian
3	1	Pengantar Teori Bilangan
4	1	Teori Grup
5	1	Teknik Penarikan Sampel
6	1	Kalkulus Dan Geometri Analisis Jilid 2
7	1	Aljabar rendah jilid III
8	1	Geometri Analitik Bidang Dan Ruang
9	1	Pengendalian Kualitas Satatistika
10	1	Belajar Pemrograman VB6 Dalam Sekejap
11	1	Statistika Dan Rancang Percobaan
12	1	Pengendalian Kualitas Satatistika
13	1	Regresi Dan Korelasi Dalam Penggandaan
14	1	Statistika Deskriptif dalam Bidang ekonomi dan Niaga
15	1	Teknik statistika untuk Bisnis & ekonomi Edisi kesembilan jilid 1
16	1	Teknik statistika untuk Bisnis & ekonomi Edisi kesembilan jilid 2
17	1	Teknik statistika untuk Bisnis & ekonomi Edisi kesembilan jilid 3

Tabel 4.23 pada pengujian *Update* yaitu mengubah nama kolom yang ada pada tabel nama_buku menjadi judul_buku

Dalam merubah nama judul kolom pada tabel 4.22 yaitu title menjadi judul buku yang tertera pada tabel 4.23. yang perlu diketahui bahwa perubahan nama kolom dapat mempengaruhi struktur pada *database*.

2. *Drop*

Drop berfungsi untuk menghapus data yang telah disimpan dalam *database*. Bedanya dengan *delete*, *drop* berfungsi menghapus secara keseluruhan beserta tabel yang ada pada *database*, Simbol (–) digunakan untuk menunjukkan operasi *Drop*.

Table 4.24 Jenis table dalam *database* Slims

table
backup_log
biblio
biblio_attachment
biblio_author
biblio_custom

Tabel 4.24 adalah jenis tabel dalam *database* slim yang di gunakan dalam menguji

Dalam percobaan Operasi *Drop* menggunakan manipulasi Aljabar Relasional table database.

Aljabar Relasional :

Biblio_author – *Biblio_author*

Berdasarkan Aljabar Relasional perintah *Drop* dapat di tuliskan dalam Sintaks SQL:

```
Drop table biblio_author;
```

Berdasarkan sintaks SQL dapat di jelaskan bahwa *Drop* tabel adalah perintah yang di gunakan untuk menghapus data-data pada tabel, *biblio_author* adalah nama tabel yang di hapus.

Table 4.25 Hasil penggunaan *Drop* pada *database*

table
backup_log
biblio
biblio_attachment
biblio_custom

Tabel 4.25 adalah tabel hasil dari pengujian *Drop* pada tabel *biblio_author*

Pada perintah *drop* bertujuan menghapus secara keseluruhan dan perintah ini mempengaruhi perubahan struktur pada *database*, yang di mana menghilangkan tupel *biblio_author* pada *database* yang tertera pada tabel 4.24 sehingga tupel *biblio_author* di *drop* seperti pada tabel 4.25. untuk memeriksa ulang apakah tabel *biblio_author* sudah terhapus `SELECT * FROM 'biblio_author'`. Maka yang akan muncul adalah error

3. Menambah tupel

Dimana dalam pengolahan ini bertujuan menambahkan tabel baru pada *database*. Simbol (+) digunakan untuk menunjukkan operasi menambahkan tabel

Tabel 4.26 jenis tabel pada *database*

table
backup_log
biblio
biblio_attachment
biblio_author
biblio_custom

Tabel 4.26 jenis-jenis tupel yang ada pada *database*

Dalam perintah Menambah tabel menggunakan manipulasi Aljabar Relasional.

Aljabar Relasional :

+biblio_aut (slims8)

Berdasarkan Aljabar Relasional perintah menambahkan tabel dapat di tuliskan dalam Sintaks SQL :

```
CREATE TABLE biblio_aut (id INT PRIMARY KEY, nama
VARCHAR(50) NOT NULL);
```

Berdasarkan sintaks SQL dapat di jelaskan bahwa create table digunakan untuk membuat tabel, biblio_aut adalah nama tabel yang dibuat, id sebagai nilai *primary key*, nama sebagai karakter.

Table 4.27 Hasil penambahan table pada *database*

table
backup_log
biblio
biblio_attachment
biblio_author
biblio_aut
biblio_custom

Tabel 4.27 adalah tabel hasil dari penambahan tabel baru yaitu biblio_aut

Dalam perintah ini menambahkan tupel baru pada struktur database yang tertera pada tabel 4.26. yang dimana menambahkan tupel bible_aut pada struktur *database* seperti yang tertera pada tabel 4.27. untuk memeriksa apakah tabel biblio_aut sudah ada dengan cara `SELECT * FROM 'biblio_author'`.

B. Pembahasan

Adapun pembahasan berdasarkan hasil penelitian di atas adalah

1. Tabel yang digunakan dalam pengolahan *database*

Pada pengolahan *database* yang di uji ada dua jenis tabel yang digunakan yaitu tabel biblio dan tabel biblio_author

2. Data *Manipulation Language* (DML)

1. Menampilkan Data

Dalam pengujian menampilkan data terdapat dua model yaitu menampilkan data secara menyeluruh dan menampilkan data secara bersyarat. Dalam menampilkan data yang perlu di perhatikan pada tabel 4.4 menghasilkan data yang sama dengan data yang di uji pada tabel 4.3 yang mengartikan data yang di tampilkan secara menyeluruh. pada tabel 4.5 menghasilkan data yang lebih sedikit yang tertera pada tabel 4.3 yang mengartikan data yang di tampilkan secara bersyarat. Dimana dalam pengujian menampilkan data terdapat operator pembandingan yang terdiri dari $=$, \neq , $<$, $>$, \leq dan \geq .

Dalam pengujian menampilkan data secara keseluruhan, dapat dituliskan

A = tabel 4.28 dimana , $n(A) = 124$ baris

B = tabel 4.29 dimana , $n(B) = 124$ baris

Jadi dapat di tuliskan bahwa pada pengujian menampilkan data secara keseluruhan $n(A) = n(B)$.

A = tabel 4.3 dimana , $n(A) = 124$ baris

B = tabel 4.5 dimana , $n(B) = 2$ baris

Jadi dapat di tuliskan bahwa pada pengujian menampilkan data secara bersyarat $n(A) \geq n(B)$.

2. *Projeksion*

Pada pengujian *projection* terdapat dua model pengujian yaitu tanpa bersyarat dan pengujian bersyarat. Dimana pada hasil pengujian *projection* tanpa perintah bersyarat tertera pada tabel 4.7 dan pengujian bersyarat tertera pada tabel 4.8 yang dimana pengujian dari tabel 4.6. dimana perintah *projection* dapat di dijabarkan dalam bentuk matematika $R = (A_1, A_2, \dots, A_n)$ Dimana R adalah relasi atau tabel yang diperoleh dari entitas dan atau relationship. Sedangkan A_1, A_2, \dots, A_n Adalah atribut ke-1, sampai dengan atribut ke-n. dimana A_1 : merupakan atribut-atribut dan datanya yang akan ditampilkan. Sedangkan R adalah satu atau lebih tabel yang digunakan. Pada sintaks bahasa SQL bentuk proyeksi dari aljabar relasional $\pi_{A_1, A_2, A_3}(R)$ adalah `SELECT A1, A2, A3 FROM R`. dapat juga di tuliskan dalam Model matematika *projection* tanpa bersyarat $R = (A_1, A_2, A_3)$ dimana $R = \text{biblio}$, $A_1 = \text{biblio_id}$, $A_2 = \text{title}$, $A_3 = \text{call_number}$, dapat dituliskan $\text{biblio} = (\text{biblio_id}, \text{title}, \text{call_number})$. Sedangkan model matematika *projection* menggunakan perintah bersyarat $R = Z(A_1, A_2, \dots, A_n)$ dimana Z merupakan perintah bersyarat $\text{call_number} = 519.4$

3. *Union*

Dimana pada hasil pengujian *Union* tertera pada tabel 4.11 yang dimana penggabungan tabel 4.9 dan tabel 4.10, dalam pengujian *union* dapat dituliskan dalam bentuk matematika $n(A) \cup n(B)$ dimana

A = tabel 4.34 dimana , $n(A) = 124$ baris

B = tabel 4.35 dimana $n(B) = 124$ baris

Dimana dalam penggabungan dua tabel dapat dituliskan dalam matematika $n(A) \cup n(B) = n(A) + n(B)$. Yang dimana $n(A)$ = tabel biblio dan $n(B)$ = tabel biblio_author. Dimana dalam menampilkan data pada perintah *union* menggunakan fungsi *outer join* yang dimana menampilkan seluruh tabel yang di gabungkan berdasar nilai biblio_id.

4. *Insert*

Dimana insert digunakan untuk menambahkan data pada suatu tabel/relasi. Jika terdapat suatu tabel/relasi r dengan suatu ekspresi E yang berisi data-data yang akan dimasukkan pada tabel maka bentuk umumnya pada persamaan $r^1 \leftarrow r^2 \cup E$. Dimana r^1 = model baru, r^2 = model lama dan E = data yang di tambahkan. Dimana r^1 = tabel 4.13 dan r^2 = tabel 4.14.

5. *Delete*

Pada pengujian *Delete* terdapat dua model pengujian yaitu tanpa bersyarat dan pengujian bersyarat. *Delete* digunakan untuk menghapus data pada suatu tabel/relasi. Jika terdapat suatu tabel atau relasi r dengan suatu kondisi ekspresi E yang akan menentukan data mana yang akan dihapus, maka

bentuk umum perintah *delete* tanpa bersyarat adalah $r^1 \leftarrow r^2 - E$. Dimana r^1 = model baru, r^2 = model lama dan E = data yang di hapus. Dimana r^1 = tabel 4.15 , r^2 = tabel 4.14. dan E adalah tabel biblio atau sama dengan tabel 4.14. Sedangkan bentuk umum perintah *delete* bersyarat adalah $r^1 \leftarrow r^2 - E$. Dimana r^1 = model baru, r^2 = model lama dan E = data yang di hapus. Dimana r^1 = tabel 4.16 , r^2 = tabel 4.14. dan E = (biblio_id = 1).

6. *Update*

Dimana yang tertera pada tabel 4.18 merupakan hasil dari pembaruan dengan perintah *update* tanpa bersyarat pada tabel 4.17. Jika terdapat tabel/relasi r dengan ekspresi perubahan data E pada suatu atribut A, maka bentuk umum *update* adalah $\delta_A \leftarrow E(r)$, dimana δ = update, A = biblio, E = title, r = analisis deret dan waktu. Dimana yang tertera pada tabel 4.19 merupakan hasil dari pembaruan dengan perintah *update* menggunakan perintah bersyarat pada tabel 4.17. Jika terdapat tabel/relasi r dengan ekspresi perubahan data E pada suatu atribut A, maka bentuk umum *update* adalah $\delta_{A(z)} \leftarrow E(r)$, dimana δ = update, A = biblio, z = (biblio_id = 1), E = title, r = analisis deret dan waktu.

3. Data Definiton Language (DDL)

1. *Rename*

Pada percobaan rename merubah nama tabel dapat di tuliskan dalam bentuk matematika $\rho_A(Z)$, dimana ρ = perintah *rename*, A = nama_buku, Z = biblio. Atau dapat dituliskan $\rho_{Newtable}(OldTable)$ adapun hasil perubahan nama tabel tertera pada tabel 4.21. Pada percobaan merubah nama kolom

dalam suatu tabel dapat di tuliskan dalam bentuk $\rho_{Newkolom}(Oldkolom)$ dimana ρ = perintah rename, A = judul_buku, Z = title, adapun hasil perubahan nama kolom tertera pada tabel 4.23.

2. *Drop*

Drop bertujuan untuk menghapus data yang ada dalam *database*, dimana dapat di tuliskan dalam model matematika. Jika terdapat suatu database P dengan suatu tabel atau relasi E yang akan menentukan data mana yang akan di *drop*, maka bentuk umumnya adalah $P \leftarrow P - E$. Dimana P = *database*, E = data.

3. Menambahkan tabel

Adapun hasil dari pengolahan tabel dengan perintah menambahkan tabel yang tertera pada tabel 4.27. dalam matematika dapat di tuliskan $P \leftarrow P + E$. Jika terdapat suatu database P dengan suatu tabel atau relasi E yang dimana akan menghasilkan data baru pada *database*. Dimana P = *slims* dan E = *biblio_aut*.

BAB V

PENUTUP

A. Kesimpulan

Penerapan *Database Query* dengan menggunakan aljabar relasional terdapat dua model yaitu *Data Manipulation Language* (DML) pengolahan database yang tidak mempengaruhi perubahan struktur database dan *Data Definiton language* (DDL) pengolahan *database* yang mempengaruhi perubahan struktur pada *database*, dimana Aljabar Relasional merupakan bahasa *query* prosedural yang terdiri dari sekumpulan operasi yang masukaannya satu atau dua relasi dan keluarannya adalah relasi baru atau dapat di tuliskan dalam bentuk matematika $RB = ((RL)(operasi))$ untuk satu Relasi, dan $.RB = ((R1)(R2))$ untuk dua relasi, dimana relasi baru (RB) dan relasi lama (RL)

B. Saran

Adapun saran pada penelitian ini adalah membuat sistem *database* pada ruang baca dengan menggunakan aplikasi slims untuk mempermudah dalam menemukan buku yang ingin di cari.

DAFTAR PUSTAKA

- A.B.Pangabean. 2008. *Kalkulus*. Yogyakarta : Graha ilmu.
- Ayres ,Frank dan Philip A. Schidmit. 2003. *Matematika Universitas*. Jakarta: Erlangga.
- Bobihu,yusran. 2013. *Pengantar Sistem Basis Data*. Makassar : Alauddin University Press.
- Darmanto, Eko. 2015 .“*Analisa Optimalisasi Bahasa SQL Berdasarkan Relational Algebra pada Kasus Rekapitulasi Mahasiswa Layak Wisuda*”. Jurnal Simetris, Vol 6 No 2.
- Fathansyah. 1999. *Database Buku Teks Ilmu Komputer*. Bandung : informasi.
- H.Rosen, Kenneth. 1998. *Discrete Mathematics and its Aplications*. New York : MCGRAW-Hill.
- Kementrian Agama RI. 2008. *Al-Quran dan Terjemahannya*. Bandung.
- Kroenke, David M. 2003.*Database Processing Jilid 1*. Jakarta : Erlangga.
- M. Shihab, Quraish. 2003. *Al-Misbah*. Jakarta: Lentera Hati.
- Mulyadi, dkk. 2017. “*Implementasi Operasi Relasional Extensible Markup Language (XML) Pada Sistem Informasi*”. Jurnal IT, Volume 8 No 1.
- Prayudi. 2006. *Kalkulus Fungsi Satu Variabel*. Yogyakarta: Graha ilmu.
- Ramakrishnan, Reghu dan Johannes gehrke. 2004. *Sistem Manajemen Database*. Yogyakarta : Andi and McGraw-Hill Education.
- Soedadyatmodjo dan Wikaria Gazali. 2005. *Kalkulus*. Yogyakarta : Graha ilmu.
- Subekti, H. Muhammad. 2004. *System Manajemen Basis Data*. Bogor. ghalia Indonesia.

Whitehorn ,Mark dan Bill Marklyn. 2001. *Seluk Beluk Database Relasional*.
Jakarta: Erlangga.



Lampiran

Tabel Biblio

biblio_id	gmd_id	title	sor	edition	publisher_id	call_number
1	1	Analisis Deret dan Waktu			0	519.4
2	1	Statistika Untuk Penelitian			0	519.4
3	1	Pengantar Teori Bilangan			0	
4	1	Teori Grup			0	
5	1	Teknik Penarikan Sampel			0	
6	1	Kalkulus Dan Geometri Analitis Jilid 2			0	
7	1	Aljabar Rendah Jilid III			0	
8	1	Geometri Analitik Bidang dan Ruang			0	
9	1	Pengendalian Kualitas Statistika			0	
10	1	Belajar Pemrograman VB6 dalam Sekejap			0	
11	1	Statistika dan Rancangan Percobaan			0	
12	1	Pengendalian Kualitas Statistika			0	
13	1	Regresi dan Korelasi dalam genggamananda			0	
14	1	Statistika deskriptif dalam bidang Ekonomi dan Nia...			0	
15	1	Teknik statistika untuk Bisnis & ekonomi Edisi kes...			0	

16	1	Teknik statistika untuk Bisnis & ekonomi Edisi kes...			0	
17	1	Teknik statistika untuk Bisnis & ekonomi Edisi kes...			0	
18	1	Statistika untuk bisnis dan ekonomi			0	
19	1	Statistika Teori dan Aplikasi Edisi Ketujuh Jilid ...			0	
20	1	Pengantar statistika matematika			0	
21	1	Statistika untuk penelitian			0	
22	1	Probabilistik dan statistik			0	
23	1	Statistik			0	519.5
24	1	Statistik untuk ekonomi dan bisnis			0	519.5
25	1	Mathematical Statistics with applications Edisi 7 ...			0	
26	1	Prinsip-prinsip Statistik untuk bisnis teknik dan ...			0	
27	1	Statistika untuk pengambilan keputusan			0	519.5
28	1	Bank soal teori statistika matematik dan penyelesa...			0	
29	1	Statistik jilid 3			0	519.5
30	1	Statistik Nonparametris			0	519.5
31	1	Metode Statistika			0	519.5
32	1	Analisis regresi Teori, Kasus, dan Solusi edisi 2			0	519.5
33	1	Statistika terapan			0	519.5

34	1	Statistika matematis lanjutan			0	519.5
35	1	Soal-jawab Statistik dengan SPSS dan Excel			0	
36	1	Statistika Deskriptif peubah banyak			0	519.5
37	1	Statistika Deskriptif peubah banyak			0	519.5
38	1	Pengantar Pengendalian kualitas statistik			0	519.5
39	1	Ilmu peluang dan statistika untuk insinyur edisi 4			0	519.5
40	1	Statistika Sebaran bebas			0	519.5
41	1	Statistik Schaum's easy outlines belajar super cep...			0	519.5
42	1	Statistika Penelitian			0	519.5
43	1	Applied Multivariate, statistical analysis edisi 6			0	
44	1	Research methods and statistics edisi third			0	
45	1	Regression Analysis by example fourth edition			0	519.5
46	1	Applied linear regression third edition			0	519.5
47	1	Computational Statistics second edition			0	519.5
48	1	Applied statistics in business & economics			0	519.5
49	1	Biostatistical Methods the Assesment of relative R...			0	519.5
50	1	Applied Multivariate analysis			0	519.5
51	1	Introduction to business statistics			0	519.5

52	1	Statistika Deskriptif dalam bidang Ekonomi dan Nia...			0	519.5
53	1	Statistika deskriptif untuk ekonomi dan bisnis			0	
54	1	Dasar-dasar ekonometrika edisi 3 jilid 1			0	
55	1	Dasar-dasar ekonometrika edisi 3 jilid 2			0	519.5
56	1	Ekonometrika			0	519.5
57	1	Statistika Matematika II			0	519.5
58	1	Pengendalian kualitas statistik (pendekatan teoritis dan aplikatif)			0	519.5
59	1	Ekonometrika deret waktu			0	519.5
60	1	Ekonometrika dasar			0	519.5
61	1	Statistika deskriptif untuk penelitian dilengkapi perhitungan manual dan aplikasi SPSS versi 17			0	519.5
62	1	Analisis multivariat arti & interpretasi			0	519.5
63	1	Pengantar statistika			0	519.5
64	1	Metode statistika nonparametrik terapan			0	519.5
65	1	Biostatistika untuk kedokteran dan kesehatan masyarakat			0	519.5
66	1	Pengendalian kualitas statistik pendekatan teoritis & Aplikatif			0	519.5
67	1	Ekonometrika			0	519.5
68	1	Analisis regresi edisi kedua			0	519.5

69	1	Rancangan percobaan aplikatif			0	519.5
70	1	Statistik pasar modal keuangan & perbankan			0	519.5
71	1	Teknik sampling untuk survey & eksperimen			0	519.5
72	1	Dasar-dasar statistika edisi ketiga			0	519.5
73	1	Pengantar statistik			0	519.5
74	1	Analisis korelasi, regresi, dan multivariate denga...			0	519.5
75	1	Statistik SIX Sigma dengan six sigma dengan MINITA...			0	519.5
76	1	Statistik jilid 3			0	519.5
77	1	Reliabilitas dan validitas edisi 4			0	
78	1	Teknik pengambilan sampel			0	519.5
79	1	C++ for mathematicians			0	
80	1	Modern database management			0	519.5
81	1	Pengantar Teknologi informasi			0	519.5
82	1	Computer Networking			0	519.5
83	1	Maple V			0	519.6
84	1	Panduan Visual Photoshop 7 untuk Fotografer			0	519.6
85	1	Computer Since			0	519.6
86	1	Dasar-dasar desain, dan implementasi database proc...			0	519.6

87	1	Seluk beluk database relasional			0	519.6
88	1	Software engineering			0	519.6
89	1	Dasar pemrograman WEB dinamis menggunakan PHP			0	519.6
90	1	Aplikasi WEB dengan PHP dan Mysql			0	519.6
91	1	Analisis data dengan exel 2007			0	519.6
92	1	Kitab suci desktop Programmer			0	519.6
93	1	Turbo pascal			0	519.6
94	1	Algoritma + Struktur Data = Program			0	519.6
95	1	Adobe photoshop CS3 untuk pemula			0	519.6
96	1	Sistem informasi Akademik Kampus berbasis WEB deng...			0	519.6
97	1	Turbo pascal			0	519.6
98	1	Turbo pascal			0	519.6
99	1	Membangun Aplikasi Toko dengan Visual Basic 2008			0	519.6
100	1	Pembuatan Program C			0	519.6
101	1	Sukses Membuat Laporan Keuangan dengan MYOB v.17			0	519.6
102	1	Logika Matematika untuk Ilmu Komputer			0	519.6
103	1	Algoritma Pemrograman			0	519.6
104	1	Jago PHP & MySQL			0	519.6

105	1	Penengenalan Umum Analisis Statistika dengan SAS			0	519.6
106	1	I.T Kreatif SPSS 18			0	519.6
107	1	Mudah & Cepat Menguasai Visual Basic			0	519.6
108	1	Buku Pintar Microsoft Excel			0	519.6
109	1	Matlab Programming			0	519.6
110	1	Smart Olah Data Penelitian dengan SPSS 21			0	519.6
111	1	MySQL untuk Pemula			0	519.6
112	1	Membangun aplikasi Smart Client dengan Visual C# d...			0	519.6
113	1	Aplikasi Excel dalam Grafik Bisnis Terapan			0	519.6
114	1	Belajar Statistika dari Nol dengan SPSS			0	519.6
115	1	Borland Delphi 5.0			0	519.6
116	1	Jomla!			0	519.6
117	1	Smart Olah Data Penelitian dengan SPSS 21			0	
118	1	Algoritma Pemrograman			0	519.6
119	1	Tips dan Trik Matlab Vektorisasi, Optimasi dan Man...			0	519.6
120	1	Praktikum Microsoft Windows Vista			0	519.6
121	1	Minitab Programming			0	519.6
122	1	Keterampilan Dasar Pengoperasian Komputer			0	519.6

123	1	123 Jurusan Mengoptimalkan Windows Vista Menjadi Mah...			0	519.6
124	1	Differential Equations DeMyStiFieD			0	519.6

Tabel Biblio Author

biblio_id	author_id	level
1	1	1
2	2	1
3	3	1
4	4	1
5	5	1
6	6	1
7	7	1
8	8	1
9	9	1
10	10	1
11	11	1
12	12	1
13	13	1
14	14	1
15	15	1
16	15	1
17	15	1
18	16	1
19	18	1
20	19	1
21	20	1
22	21	1
23	21	1
24	16	1
25	22	1
26	23	1

27	24	1
28	19	1
29	25	1
30	20	1
31	26	1
32	27	1
33	28	1
34	19	1
35	29	1
36	30	1
37	30	1
38	31	1
39	17	1
40	1	1
41	21	1
42	32	1
43	33	1
44	34	1
45	35	1
46	36	1
47	37	1
48	38	1
49	39	1
50	40	1
51	41	1
52	14	1
53	42	1
54	43	1
55	43	1
56	44	1
69	54	1
68	53	1
70	18	1
71	18	1
72	1	1
73	55	1
74	56	1
75	57	1



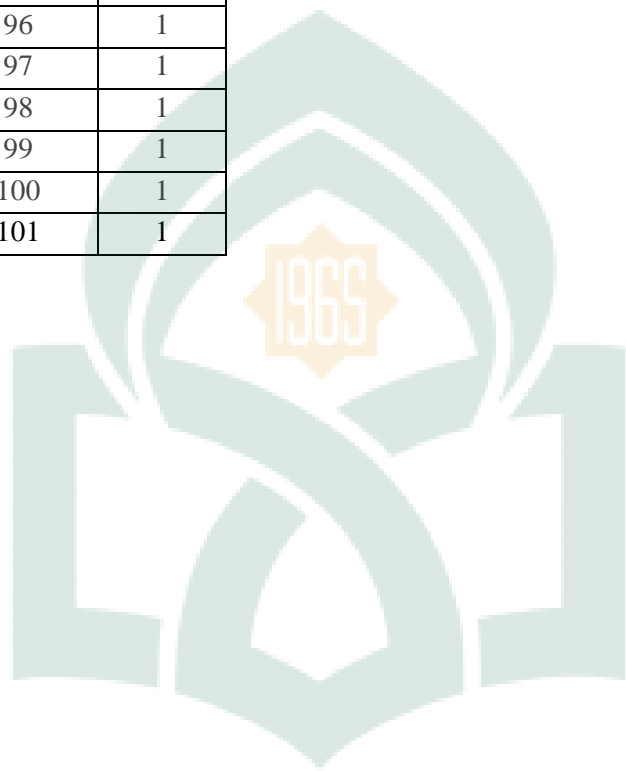
UNIVERSITAS ISLAM NEGERI
ALAUDDIN
 MAKASSAR

76	25	1
77	58	1
78	59	1
79	60	1
80	61	1
81	62	1
82	63	1
83	64	1
84	65	1
85	66	1
86	67	1
87	68	1
88	69	1
89	70	1
90	71	1
91	72	1
92	73	1
93	74	1
94	75	1
95	76	1
96	77	1
97	78	1
98	74	1
99	79	1
100	80	1
101	81	1
102	82	1
103	83	1
104	84	1
105	85	1
106	86	1
107	87	1
108	88	1
109	89	1
110	90	1
111	91	1
112	92	1
113	72	1



UNIVERSITAS ISLAM NEGERI
ALAUDDIN
 MAKASSAR

114	93	1
115	94	1
116	95	1
117	90	1
118	83	1
119	96	1
120	97	1
121	98	1
122	99	1
123	100	1
124	101	1



UNIVERSITAS ISLAM NEGERI
ALAUDDIN
 M A K A S S A R

RIWAYAT HIDUP



Firmansyah Salam lahir di Makassar 06 september 1996 dari seorang ibu bernama Nursyam dan Ayah bernama Salam Nur. Pendidikan penulis SD Inpres Bangkala 1, SMP Negeri 19 Makassar, SMA Negeri 13 Makassar, dan Universitas Islam Negeri Alauddin Makassar. Penulis sekarang tinggal di Jl Muhajirin Raya No 27 Kelurahan Bangkala , Kecamatan Manggala Kota Makassar Sulawesi Selatan.

UNIVERSITAS ISLAM NEGERI
ALAUDDIN
M A K A S S A R